UNIVERSITY
OF FLORIDA
LIBRARIES

COLLEGE LIBRARY

Digitized by the Internet Archive
in 2011 with funding from
LYRASIS Members and Sloan Foundation

http://www.archive.org/details/truth00mart

TRUTH AND DENOTATION

TRUTH & DENOTATION

A Study in Semantical Theory

by

R. M. MARTIN

THE UNIVERSITY OF CHICAGO PRESS

Library of Congress Catalog Number : 57-12813

THE UNIVERSITY OF CHICAGO PRESS, CHICAGO 37

The University of Toronto Press, Toronto 5, Canada

Routledge & Kegan Paul Ltd., London, E.C.4, England

Copyright under the International Copyright Union, 1958. All rights reserved. Published 1958. Printed in Great Britain by Butler & Tanner Ltd., Frome and London

© *1958 by Richard Martin*

To the memory of

ALFRED NORTH WHITEHEAD

the scientist,
the philosopher,
the human being,
the teacher,
the friend.

Trace science then, with modesty thy guide:
First strip off all her equipage of pride;
Deduct what is but vanity or dress,
Or learning's luxury, or idleness;
Or tricks to show the stretch of human brain,
Mere curious pleasure, or ingenious pain;
Expunge the whole, or lop th' excrescent parts
Of all our vices have created arts;
Then see how little the remaining sum,
Which serv'd the past, and must the times to come!

POPE, *Essay on Man*, II, 1

PREFACE

EVERY reflective student of language is interested in how predicates and names are related to each other and to the objects for which they stand. The study of such *semantical* topics has played an important role in philosophy and grammar from the Greek period to the present day. Until recently philosophical discussion of them has been intimately bound up with more general metaphysical or epistemological considerations, and semantical theory has been a branch of a broader philosophy of language. But today interest seems to focus on such theory independent of its place in any metaphysical or epistemological scheme. Semantics is now concerned with problems of a narrower kind than heretofore, but these have come to occupy a central place in contemporary philosophical discussion.

Within recent years semantical theory has been tremendously revitalized by discoveries in modern symbolic logic, discoveries so fundamental that semantics has become virtually a branch of logic. Most of the important contributions to modern logic including semantics have been of a highly technical kind, and many of them have been made by professional mathematicians. Theoretical semantics might thus with justice be called *mathematical* semantics. But to call it such would suggest that the subject is of narrow interest and relevant only to studies in the foundations of mathematics. On the contrary, the kinds of issues with which it deals are of wide interest and are important for studies in the foundations of theoretical, systematic disciplines of all kinds. In particular, the methodology of the sciences is now pre-eminently concerned with the semantics of scientific language. Also the basic concepts of semantics are of interest for philosophy in its various branches.

This book is a study in the logical foundations of modern theoretical semantics and is concerned with such notions as truth, denotation, designation, consistency, and the like. Several alternative semantical theories are to be formulated, and most of them contain some novelty, however slight, of conception or of treatment. Some of the theories formulated are perhaps new, some of them are old, and some of them are old theories formulated in a somewhat new way. No specific knowledge is presupposed on the part of the reader other than some familiarity with the most elementary logic of truth-functions and quantifiers. Mathematical complexities are at a minimum, and indeed one will find in the semantical material of this book only the

simplest mathematics. On philosophical grounds this is thought to constitute a merit rather than a defect, and will perhaps help to make some of the important concepts and methods of modern semantics available to a wider audience.

Under semantics are often included theories concerned with the analysis of meaning at the common-sense level, with meaning and expression in literature and the arts, and even with schemes aimed at the betterment of the world's social, political, and economic ills! These kinds of theories, some of them of genuine interest, usually rest upon unanalyzed foundations. Their root ideas or concepts are often not clear, nor have their basic assumptions undergone logical analysis. Such theories seem to presuppose something more basic by way of logical substructure. There is urgent need for an approach to problems of a general semantical nature from a secure theoretical basis. But the semantics of this book is concerned solely with denotation and allied notions, and the discussion is therefore somewhat narrower, more technical, and more exact than is usual in philosophical treatises of a more general kind.

Often semantics is taken as containing also a theory of meaning or of intension. The intension of a term is then distinguished from the object for which it stands somewhat as, traditionally, the connotation of a term has been distinguished from its denotation. In this book we shall be occupied only with semantical theories of extension or denotation, which are usually presupposed by, or contained in, systematic theories of intension in one way or another. Even if we confine ourselves to theories of denotation or extension, we have a subject of great richness and fascination. This becomes especially evident if one reflects upon the intimate connection between denotation and the semantical concept of truth. One of the important achievements of modern semantics is to have provided a rigorous, logico-mathematical analysis of the truth-concept of formalized language-systems of certain kinds. Truth, denotation, and other semantical notions are closely interrelated. A denotational semantics of sufficient power must contain a semantical truth-concept also.

The theories here are to be systematic and concerned only with the semantics of formalized language-systems of a certain kind. The treatment is systematic or formal in the sense that the semantical theories themselves will be presented within formalized language-systems, the full structure of which will be specified. The so-called "object"-languages, the languages whose semantics is under discussion, will as is customary have their syntactical and semantical structure fully described. The so-called semantical "meta"-languages, the languages in which the semantical theory of a given object-

language is couched, will also be formulated here with care as formal deductive systems. In this way we shall gain maximal clarity and explicitness in the presentation of the various theories.

A fairly full and comprehensive exposition of the formal material is given throughout, and many theorems are stated here and there. The proofs will often be omitted, but some will be sketched, at least to the extent of giving the nerve of the argument. Enough will be given so that those who are familiar with proofs on the basis of modern logic will be convinced without having to go through the detailed labor of proof; yet those who are not so skilled will not be deprived the pleasure of constructing rigorous proofs for themselves.

Because theoretical semantics is concerned primarily with problems of language and the presentation of the material here presupposes only the most elementary logic, the book will perhaps be useful to readers and students in such fields as logic, analytical philosophy, epistemology, metaphysics, linguistics, philosophy of language, and the philosophy of science. Some of the material covered has been presented in a graduate logic seminar at the University of Pennsylvania during the past few years, to which qualified undergraduates were also admitted.

On every page of this book the influence of the syntactical and semantical writings of Carnap, Church, Frege, Goodman, Quine, Russell, and Tarski will, it is hoped, be evident. Without the work of these authors, who have helped to develop modern syntax and semantics, the theories here could not have been formulated. To enumerate in detail specific items of indebtedness would seem therefore somewhat gratuitous. However, to the writings of Carnap, Church, Goodman, Quine, and Tarski the author is especially indebted for technical details, and to those of Carnap, Goodman, and Quine for some features of the general philosophical outlook.

Some of the theories presented are formulated here for the first time and may no doubt be improved upon in various respects. Perhaps some of them can be simplified, generalized, or made more adequate in this way or that. Further, this book is not the handiwork of a professional mathematician, and the indulgence of the mathematical reader is therefore asked for such oversights, errors, or lapses from rigor as he may find. Where specific statements can be shown false, this will at least indicate (in the comforting phrase of Professor H. H. Price) that they are not meaningless! Some of the metalanguages formulated here may be of minor mathematical interest, but this is not claimed. The motivation is wholly philosophical, as will be evident throughout.

The author wishes to express his appreciation to Professor Carnap

for several valuable criticisms and suggestions. Also the author wishes to thank Professor J. H. Woodger for some basic conceptions and for many fruitful discussions. In a sense this book is due primarily to Professor Woodger, because the study of some of the semantical theories developed here was first undertaken at his suggestion. The author is also indebted in various ways to Professors Fitch, Hempel, and Nagel, who have influenced for the better (but in ways perhaps unknown to them and of which they might not approve) the material of this book. The author is indebted in other respects to Professors Beth and Hiz and to his colleagues Professors Clarke, Morrow, and Schrecker. Thanks are due to Mr. John Beierle, Jr., Miss Ruby Meager, Mr. G. F. Rieman, Jr., Mrs. Beverly Levin Robbins, and Mr. T. R. Savage for expert editorial and other aid in the preparation of the manuscript.

The author wishes also to express his appreciation to the John Simon Guggenheim Memorial Foundation and the University of Pennsylvania for freedom from academic duties during the session 1951–2, when the first draft of this book was written.

Grateful acknowledgments are made herewith to the Cambridge University Press, the University of Chicago Press, the Dial Press, Harvard University Press, and the American Mathematical Society for kind permission to quote from works published by them.

Finally, thanks are due to the Research Committee of the University of Pennsylvania for grants covering clerical aid.

CONTENTS

PREFACE vii

I. PHILOSOPHICAL ANALYSIS AND FORMALIZED LANGUAGES 1

A. Formalization—B. Interpretation—C. Chief Merits of Formalization and Interpretation—D. Mathematical, Scientific, and Philosophical Languages—E. Explication and Rational Reconstruction—F. Paradoxes, Natural Language, and Approximation—G. Minimal Core of Semantical Theory—H. Summary.

II. FIRST-ORDER LANGUAGES 31

A. Logical Systems of First Order—B. Some Truth-Functional Theorems—C. Theorems of Quantification—D. Abstraction—E. Identity—F. Singular and Selective Descriptions—G. Normal Forms—H. Domains of Individuals—I. Object-Language and Meta-Language—J. Syntax and Semantics—K. Sign-Designs and Sign-Events—L. Some First-Order Object-Languages.

III. FIRST-ORDER SYNTAX 70

A. Primitives—B. Rules—C. Some Elementary Properties of Concatenation—D. Some Preliminary Definitions—E. 'Axiom of *L*'—F. 'Theorem of *L*'—G. Some Syntactical Theorems—H. Some Further Notions of Syntax.

IV. MULTIPLE DENOTATION: FIRST-ORDER SEMANTICS 99

A. Multiple Denotation—B. Primitive Framework—C. Some Definitions—D. Rules of Semantics—E. Some Elementary Theorems—F. Boolean Algebra of Predicate Constants.

V. TRUTH, ADEQUACY, AND CONSISTENCY 115

A. Truth—B. Some Alternative Definitions—C. Proof of Adequacy—D. Correspondence—E. Semantical Consistency of *L*—F. Some Further Notions of Semantics—G. Relative Consistency of SM_D^L—H. Self-Reference—I. Undecidability and Truth.

VI. SET THEORY AND THEORY OF TYPES 143

A. Zermelo-Skolem System *S*—B. Definition of 'Axiom of *S*' within SM_D^S—C. Theory of Types—D. 'Axiom of *T*' within SM_D^T—E. Semantical Notions within SM_D^T—F. Re-Interpretation—G. Language-Systems of Finite Order.

VII. SATISFACTION, DESIGNATION, DETERMINATION 165

A. Satisfaction—B. Designation in T—C. Determination in T—D. Concerning Designation and Determination in S—E. The Definition of Semantical Relations—F. In Favor of SM_D^L and Allied Semantical Meta-Languages.

VIII. NON-TRANSLATIONAL SEMANTICS (I) 179

A. Primitives — B. Some Definitions — C. Truth — D. General Rules of Comprehension—E. Some Alternative Bases.

IX. NON-TRANSLATIONAL SEMANTICS (II) 191

A. Boolean Algebra — B. Some Further Theorems — C. Semantical Implexion of Predicate Constants—D. Logical Theorems and Truth—E. Specific Hypotheses of Comprehension—F. Set-Theoretical Theorems and Truth—G. Relative Consistency of Non-Translational Semantics.

X. INDIVIDUAL AND FUNCTIONAL CONSTANTS 213

A. Syntactical Readjustments—B. Individual Constants and Multiple Denotation—C. Relational Abstraction—D. Individual Constants and Comprehension—E. Some Semantical Theorems—F. Functional Constants—G. Self-Reference and Comprehension.

XI. INSCRIPTIONS AND MULTIPLE DENOTATION 227

A. Syntactical Preliminaries — B. Some Elementary Theorems—C. Definition of 'Fmla'—D. Definition of 'Thm'—E. Some Further Theorems—F. Multiple Denotation and Truth—G. Adequacy—H. Truth and Vacuous Abstracts.

XII. INSCRIPTIONS AND COMPREHENSION 254

A. Semantical Preliminaries—B. Some Theorems—C. Truth and Derivative Notions—D. Consistency and Relative Consistency—E. Individual Constants and Inscriptions—F. Formal Theory of Quotation Marks—G. Semantics Based upon Inscriptions and Shapes.

XIII. FIRST-ORDER CONSTRUCTIVISM 263

A. The Purport of the Maxim—B. Consistency and Economy—C. Semantic Constructivism, Nominalism, Realism, Finitism—D. Classes—E. In Favor of Non-Translational Semantics—F. Minimality—G. On Understanding a Language-System—H. Toward a Formal Theory of Semantical Systems—I. Paradox of Skolem.

INDICES 296

CHAPTER I

PHILOSOPHICAL ANALYSIS AND FORMALIZED LANGUAGES

An important and fruitful approach to the problems of contemporary philosophy and methodology of science is by means of the study of formalized language-systems. A formalized language is an artificial thing constructed self-consciously by the working methodologist. It is perhaps not strictly a language at all, in the ordinary sense of the word. A formalized language consists rather of a precise set of rules and definitions explicitly laid down for given purposes. There is often considerable choice as to the rules and definitions and many details in the formulation of such languages are matters of convention. Bearing in mind specific purposes one constructs or chooses a language-system explicitly to serve those purposes. If one's aim is to formalize the elementary arithmetic of positive integers in a certain way the language-system is chosen accordingly. If the aim is rather to formulate the laws governing mitosis in the common fruit-fly, *Drosophila melanogaster*, the language is constructed with this end in mind. Or, if one wishes to formulate a metaphysical or epistemological theory of the relations, say, between objects of sensation and physical objects, one chooses an appropriate language-system containing expressions for the kinds of relations and objects in question.

Formalized languages are thus to be contrasted sharply with the so-called "natural" languages. By a natural or ordinary language is usually meant one of everyday speech, such as English, Swahili, or Hindustani. Such natural languages are the most used media of communication in practical affairs. Our science, literature, technology, etc., are usually expressed in them as well as our common-sense judgments. And the analysis and clarification of the structure of natural language is a task of great philosophical interest.

The approach to philosophical and methodological problems by means of the study of formalized languages, however, is thought to possess some important advantages over other methods. These are essentially the advantages which the refined tools of the exact sciences have over pre-systematic, rough, classificatory procedures. Precisely what are these advantages? What are formalized languages? What is to be achieved by using them? How are they related to

theoretical semantics? How can the use of formalized languages be helpful in the analysis of natural language? These questions we attempt to answer to some extent in the present chapter.[1]

In § A a rough sketch of the procedure of formalizing a language is given, and the (semantical) method of providing an *interpretation* for it is discussed in § B. The *chief merits* of the methods of formalization and interpretation are enumerated in some detail in § C. In § D the role of these methods within *science* and *philosophical analysis* is indicated roughly. The connection of formalization with "*rational reconstruction*" and "*explication*" is brought out in § E. An important argument in favor of formalized languages arises from considerations concerning the logical and semantical *antinomies*. These are discussed informally in § F, and the *approximative approach* to natural languages by means of very powerful formalized languages is indicated. In § G, the main purpose of the book is stated, namely, to formulate a kind of *minimal core* of semantical theory upon which further philosophical or methodological constructions may be securely based.

A. Formalization. An exact or rigorous definition of the notion of being a *language-system* is not needed for present purposes. The following rough account will suffice.[2]

Strictly speaking, we should distinguish between a formalized *logistic system* (or calculus) and a formalized *language-system* (or interpreted language) as follows. The former is determined by grammatical rules or definitions which refer exclusively to symbols and expressions, regarded in abstraction from any specific interpretation. A language-system, on the other hand, is a logistic system with a fixed, determinate interpretation given to certain of its expressions. For the moment, let us speak only of calculi or logistic systems.

Roughly speaking, a calculus or formalized logistic system consists of the following:

(i) a complete specification of the *primitive vocabulary*,

(ii) a definition (recursive or otherwise) of the notion of being a *formula*, and possibly *term*, of that system,

[1] For recent or forthcoming discussions of this same topic see, e.g., E. W. Beth, "Carnap's Views on Constructed Systems vs. Natural Languages in Analytic Philosophy", *The Philosophy of Rudolf Carnap* (*The Library of Living Philosophers*, to appear); E. Hutten, "Natural and Scientific Language", *Philosophy*, xxix, No. 108 (1954), pp. 27–43; R. B. Braithwaite, *Scientific Explanation* (Cambridge University Press, Cambridge: 1953), especially Chap. I; H. Wang, "On Formalization", *Mind*, LXIV (1955), pp. 226–38; and J. H. Woodger, "Mathematical Logic as a Tool of Analysis", to appear.

[2] See Chapter XIII, § H below.

(iii) a list of formulae as *axioms* or *primitive formulae*,

(iv) some statements about the system, *rules of inference*, stating the circumstances under which a formula is to be regarded as an *immediate consequence of* or as *immediately provable from* a formula or formulae,

(v) a list of formulae explicitly shown to be *theorems* or provable from the axioms by means of the rules of inference,

and

(vi) a list of statements about the system allowing us to *abbreviate* expressions in specified ways.

This rough description is in no way intended as an exact definition, and there are perhaps some systems which might be called 'formalized' to which it does not apply. But throughout this book we shall be talking only about systems of a certain specified kind (see Chapter II), to which it does apply. And most of the important formalized systems in the literature of modern logic and methodology are of this kind also.

Let us consider in a little more detail the various items constituting a formalized system. First we must have a specification *in toto* of the primitive vocabulary. The *expressions* of a system consist of certain signs or symbols together with all possible finite sequences of them. The signs or symbols used are the *primitive* signs. There may be a finite or denumerably infinite number of them. Any result of placing two or more primitive signs in a consecutive, linear, usually left to right, order is a *sequence* or a compound expression. The expressions can then be described as the primitive signs together with all sequences or compounds of them. Each sequence is of finite length, although the totality of sequences is usually presumed to be denumerably infinite. (Some systems have been proposed which contain even expressions of infinite length.)

The primitive signs are usually of two kinds, *constants* or *variables*. Constants are so called because what they stand for or designate remain constant throughout the given discourse. Variables, on the other hand, do not properly denote or designate; rather they are said to *range* over a certain domain of objects. Some systems contain several kinds of variables, some even an infinite number of different kinds of variables, each kind ranging appropriately over a certain domain of objects. Usually there is an infinite number of variables of each kind, although this is not strictly essential. The role which variables play in the languages to be discussed is of the greatest importance. It is by means of them primarily that one is able to achieve generality in discourse, that one is able to talk about *all* objects, *some* objects, *no* objects, etc. All the systems to be discussed

are such as to contain variables, although very powerful language-systems have been constructed which do not contain them at all.

The constants are also often of different kinds. Some are *logical* constants, some are *predicate* constants, some *individual* constants, and some *functional* constants or *functors*. The characteristics of these different kinds of constants will be specified below. (See Chapter II.)

Although any finite sequence of the primitive signs is an expression of the system, only some sequences are regarded as *formulae*. The formulae are also sometimes called *sentential functions*. The formulae may or may not be of the kind to which one would ascribe truth or falsity. If they are, they are called *sentences*, otherwise *proper sentential functions*. E.g., 'x is a Quaker', where 'x' is a variable, may be a formula in some appropriate system, but is neither true nor false in that system unless we specify something more about 'x'. The proper sentential functions of a system are such that they become sentences (and hence true or false) in ways easily described. To return to our example, 'x is a Quaker' becomes a true sentence in that system if in place of 'x' we put in the proper name 'Rufus Jones'; it becomes a false sentence if we prefix to it the quantifying phrase 'for all x' and think of 'x' as ranging over all human beings.

The notion of being a formula is often defined by means of a so-called *recursive* procedure. The definition is first given for the simplest cases (*atomic* formulae), and then, assuming it to be given for cases of a certain complexity, the definition is given for cases of greater complexity which in some sense presuppose the definition or definitions for the less complex ones. The definitions for the more complex cases presuppose or refer back to or recur to the definitions for the simpler ones. In this way all possible cases are covered. In defining the notion of being a formula, e.g., we first stipulate what the simplest (atomic) formulae are, and then stipulate how the longer (*molecular*) formulae are built out of shorter ones. This type of definition is very useful in formulating language-systems and we shall frequently employ it informally in later chapters.

Some sequences of the primitive expressions of the system are called *terms*. The terms are expressions which, either by themselves or when altered in specifiable ways, can be regarded as referring to objects. A variable by itself is a term, and any individual constant is a term. Also appropriate expressions containing functional constants (functors) are terms. In an arithmetical language, e.g., where '$+$' is a primitive functor, an expression such as '$(x + y)$' may be a term. If in place of 'x' and 'y' here we put in names of specific numbers, we gain a name for the sum of those numbers.

The complete specification of the primitive vocabulary is the first

requirement in formalizing a language, and the rigorous definition of 'formula' and possibly 'term' is the second. The third is the listing of certain formulae or sentences as axioms or primitive formulae, of which there may be either a finite or a denumerably infinite number. In practice one usually chooses as axioms only those formulae which, in an intended interpretation, are valid or true. Also, one usually prefers as axioms statements of considerable deductive power. Sometimes the axioms are also called *postulates* or *assumptions*.

Suppose we are given a complete specification of the axioms. Strictly we can *prove* nothing from them unless *rules of inference* are also available. The rules of inference are explicit statements, not *in* the system but *about* it, which state that from a formula or formulae of the system of such and such a form or forms, another formula can be inferred. The rules of inference, in other words, usually state that from formulae $\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_n$, a certain formula $\boldsymbol{B}$ can be inferred. The form of $\boldsymbol{B}$ is usually closely related to or determined by those of $\boldsymbol{A}_1, \ldots, \boldsymbol{A}_n$. Often, especially in older writings, the rules of inference are not explicitly stated. And in fact the clear-cut distinction between the notion of an axiom and that of a rule of inference is a very recent one. It is not drawn clearly, e.g., even in *Principia Mathematica*.[1] However, a moment's reflection is enough to convince one that the distinction is fundamental. Usually only two or three rules of inference are required and these are ordinarily of a very simple kind. (See Chapter II, § A.)

An important restriction upon the choice of rules of inference is that they should be such as to preserve truth in the intended interpretation of the system. In other words, starting with true sentences, the sentences which can be inferred from them by means of the rules must then also be true. A rule of inference which lacked this property would clearly not be suitable, because it would enable us to infer falsehoods from truths.

The main purpose for formulating axioms and rules of inference is to be able to deduce or prove theorems. To *prove* a formula $\boldsymbol{A}$ is, roughly speaking, to exhibit a finite sequence of formulae of which $\boldsymbol{A}$ is the last, each formula of which is either an axiom or can be reached from preceding formulae in the sequence by means of a rule of inference. The *theorems* of the system are then, by definition, the formulae for which there are proofs. If the axioms are true sentences, the rules of inference being such as to preserve truth, the theorems will then also be true. Often one can interchange some of the theorems with the axioms, and the choice as to which

[1] A. N. Whitehead and B. Russell, *Principia Mathematica* (Cambridge University Press, Cambridge: 1910–13 (1st ed.), 1925–7 (2nd ed.)).

formulae are to be axioms and which theorems is to some extent arbitrary.

Finally, it is often convenient to abbreviate the longer expressions of the system in explicitly specified ways. The statements about the system which stipulate these abbreviations are usually called *definitions*. In the strict sense definitions do nothing but abbreviate; they merely allow us to write shorter sequences of symbols in place of longer ones. Such abbreviations are not strictly essential in the development of the system. We could dispense with them altogether, although in practice they immensely facilitate discourse both in and about the system. The expression being defined is usually called the *definiendum*, and the expression by means of which it is defined, the *definiens*.

B. Interpretation. In the preceding section we were concerned wholly with syntactical or notational matters. We talked exclusively of the symbols and expressions of the system, and not about the objects which the expressions are supposed to refer to or denote or stand for in one way or another. When one speaks in this purely *syntactical* way, abstracting from the meaning or denotation of the expressions, one is of course merely postponing some of the most vital and interesting problems in the analysis or construction of language. But it seems better first to gain clarity concerning the purely syntactical character of a language before going on to a discussion of denotation or meaning. To jump *in medias res* of the difficult problems of meaning with no secure syntactical foundation would seem hazardous. Nearly everyone would no doubt assent to this in principle, but it is often forgotten in practice. Syntax is presupposed by semantics, as we shall see in the course of our inquiry. Hence clarity concerning the syntactical properties of a language-system must be gained before its semantical properties can be studied with any care.

Given a formalized logistic system there are ordinarily several different interpretations we can attach to it. An interpretation, roughly speaking, is a systematic assignment of denotata to some of the primitive vocabulary. The individual constants are taken as standing for certain well-specified entities. Usually distinct individual constants are taken as standing for distinct entities, although this is not essential. The predicate constants are often described as standing for well-specified properties of or classes of or relations between or among entities. But not all the primitive symbols are given denotata in this way. E.g., suppose the language contains parentheses. Parentheses play the role simply of punctuation marks and do not

properly denote. Also such so-called logical constants as 'and', 'or', 'if and only if', etc., need not be regarded as denoting. Of course, such words have an important role to play, as we shall see, but this role is not strictly that of denoting.

We have been using the words 'meaning' and 'denotation' very informally and as approximate synonyms. Often a distinction is made between meaning and denotation somewhat akin to the traditional distinction between connotation and denotation, or between the intension and extension of terms. 'Meaning' is then usually taken in the sense roughly of 'connotation'. The word 'meaning' will not be used hereafter as a technical term—nor indeed has it been so used above. Our concern for the present is exclusively with denotation or reference in the extensional sense.

Formalized logistic systems in themselves are not usually of interest. They become so only when appropriately interpreted. Semantics is the study of the way in which an interpretation is explicitly attached to a formalized logistic system. In the theories to be developed here this is accomplished by means of *semantical rules* or *axioms*. Semantics as conceived here is therefore primarily the study of semantical rules.

In § A we noted six items roughly constituting a logistic system. If to these we add

(vii) a list of semantical rules or axioms,

we gain an explicit interpretation. A formalized logistic system with an interpretation specified is also called a *semantical system* or an *interpreted system*.

Usually an uninterpreted logistic system can be given several alternative interpretations. All of these will, of course, have something in common; they will be alike in certain respects. But also there may be important differences. Usually one interpretation will be the basic or intended or *standard* one, the other interpretations then being *non-standard*. The study of the non-standard interpretations of certain mathematical systems has turned out to be one of the most fascinating chapters in mathematical semantics. But the general pattern of giving a non-standard interpretation to a logistic system is essentially that of giving the standard one. And because the object-languages concerned in the study of non-standard interpretations are primarily mathematical languages, we shall have little to say here specifically concerning them. (See, however, Chapter XIII, § I.)

Within formalized language-systems (in a given interpretation) one speaks of certain objects, ascribing or denying properties of

them, saying that certain relations hold among them or not, etc. These objects, together with the properties or relations, might be loosely called the *subject-matter* of the system in that interpretation. The subject-matter may vary with different language-systems as well as with different interpretations of the same underlying logistic system. Some languages have as fundamental objects neutrons, protons, mesons, etc., together with relations between or among them, e.g., a language for certain areas of nuclear physics. Others might have human beings and various relations among them as fundamental objects, e.g., a sociological language. Still another might have as fundamental objects the musical compositions of Mozart, e.g., a formalized language containing an exact analysis of the structure of Mozart's works. In principle any clear-cut subject-matter at all can give rise to a formalized language. Of course the formulation of some such systems would be simple or straightforward, that of others, very complex and difficult. Some systems might be unimportant or uninteresting for either science, philosophy, or daily affairs, whereas others might be concerned with important areas of human knowledge.

It has been remarked that semantics, in the sense in which the word is being used here, is the study of the way in which formalized logistic systems are given explicit interpretation. The argument, therefore, to be put forward concerning the importance for methodology and philosophical analysis of formalized language-systems is also an argument concerning the importance of semantics. In some philosophical circles there is little interest in modern theoretical semantics. Some methodologists doubt its relevance for the philosophy of science. Many analytic philosophers contend that semantics has little to contribute to the direct analysis of natural language. But if the arguments to be given below concerning the over-all importance of formalized language-systems are well founded, the methods of modern theoretical semantics will be seen to be of greater philosophical and methodological significance than is generally recognized.

C. Chief Merits of Formalization and Interpretation. We have now gained a rough idea as to what formalized language-systems are and what is involved in providing an interpretation for them. What do formalized languages with specified interpretations have to contribute to philosophical analysis and to the methodology of the sciences?

The first and most obvious item to point out is the clarity and precision which results from the actual procedures of formalization and

interpretation. In formalizing a language we give in effect a careful logical analysis of the subject-matter of that language. In choosing the primitives, in making precise their denotation, in defining other expressions in terms of them, in explicitly laying down the basic assumptions, etc., a clarity and precision is gained that is perhaps not possible in any other way. That this is the case is well attested to by those who have made extensive use of these procedures and denied only by those who have not.

Also the method of formalization provides an aid to intuition in any area to which it is applied. Indeed, the aid to intuition is so great that many feel that this is the supreme argument in behalf of formalized languages. Often our most basic insights or intuitions in a given area turn out to be unreliable. Especially perhaps when a subject-matter becomes more and more complex, as in a well-developed science or in a metaphysical system, a point is reached where unaided intuition cannot cope with the problems or situations presented. The presence of the Russell and other paradoxes in the mathematical theory of sets, e.g., can be construed as showing that our naïve intuition concerning the relation of membership between sets is misleading. (See Chapter VI, § C below.) If we trust solely to intuition in such matters we are led into contradiction. In formalizing a given subject-matter our intuitions are sharpened and perhaps corrected where such correction is needed. The method cuts below intuition, as it were, aiding it and abetting it by careful scrutiny of the primitives and of the alternative possible axioms and definitions.

That systematic inference within a formalized language-system is a genuine corrective to faulty thinking is also to be noted. By incorporating a scientific, methodological, or philosophical phrase or term within a well-knit systematic structure, we minimize the liability of error and protect ourselves from careless or slip-shod ways of thinking. Also this method affords a valuable critique of terms or notions the exact character of which is in doubt. The kind of acute criticism a term undergoes by incorporating it within a formalized language is an invaluable aid to its analysis. Woodger has remarked that "it is a good practical maxim, when confronted with a word about the scientific [or philosophical] respectability of which we are in doubt, to ask: What is its extension? What is its degree? [I.e., is it an individual term? A class term? A dyadic relational term? If not, what?] If a clear answer is not forthcoming the word should be regarded with suspicion and used with caution."[1]

To lay down *all* one's assumptions in a given area of knowledge

[1] J. H. Woodger, *Biology and Language*, Tarner Lectures (Cambridge University Press, Cambridge: 1952), p. 26.

or in a given context of inquiry is a simple requirement to state in principle, but often a very arduous one to follow in practice. Often more assumptions or axioms are needed than seem required at first blush. And often these are difficult to state; they may elude our intuitive grasp or be otherwise difficult to formulate. To have a full specification of the principles or laws used or presupposed in a given science or systematic discourse is a clear desideratum. The formalization of a language enables us to achieve this explicitly.

Another merit of the method of formalization is that it provides an efficient way of eliciting the *consequences* of our assumptions. We are often led intuitively to think that such and such a formula follows from the axioms to find on more careful analysis that in fact it does not. Also the examination of the consequences of the assumptions enables us better to grasp their full significance. Often unsuspected consequences turn up, perhaps even undesirable ones. In this event some change is usually required in the axioms or definitions. The *only* method we have of drawing out explicitly the consequences of the axioms is by means of deductive rules of inference within the confines of a formalized language-system.

In laying down an interpretation for a formalized system by explicitly stating the semantical rules, we employ another language, the so-called *meta-language*. The meta-language we are presumed to understand. The semantical rules are couched in the meta-language, stating specifically that appropriate expressions of the given language-system denote such and such objects. Presupposing that we understand the meta-language, we come also to understand the object-language. The procedure of interpretation is made explicit in this way. (Cf. Chapter XIII, §§ E and G.)

Also it is by reference to formalized language-systems that the important distinction between object- and meta-language itself can be clearly maintained. This distinction is of great significance and is difficult if not impossible to draw in any other terms. Its methodological importance, now established beyond doubt, constitutes another argument in behalf of formalized languages. (This will be discussed more fully in § I of Chapter II and throughout.)

Still another important distinction, already alluded to, is that between syntax and semantics. This is no doubt best drawn in terms of the different kinds of meta-languages needed for given purposes. (These will be discussed more fully in Chapter II, § J, and throughout.)

Also formalized languages provide a *basis for meta-theoretic investigations*. They provide, in other words, manageable language-structures which can be investigated *as a whole*. Studies of this kind

seem to be primarily of two sorts, those concerned with the meta-theoretic properties of a single system and those concerned with the interrelations of two or more systems. Much work of the first kind has been done, and several important languages have been subjected to detailed meta-theoretic study. But the problems in the comparative study of languages have scarcely been scratched, except perhaps in mathematics. Studies of this comparative kind would be concerned with the interrelations of the sciences or of the various areas of learning, with the reduction of one science to another, with the comparison of one philosophical system with another, and so on.

One widely studied meta-theoretical property is that of *consistency*. In the sciences and in philosophy some even very approximate or rough assurance that the language-system being used is consistent (in the sense of containing no contradiction) seems essential. Science, broadly conceived, may be regarded as an activity of sifting out, from the totality of sentences concerned with a given subject-matter, those sentences which are true. But the point of such activity is lost unless the language-system in which these sentences are couched is consistent. For if the language is inconsistent, one can show, using only clearly acceptable principles, that the extension of 'true' embraces all the sentences of the language whatsoever (see Chapter V, § E), and the distinction between truth and falsity vanishes!

The problem of proving that certain formalized languages are consistent has given rise to an important kind of meta-theoretical investigation. In fact, the consistency problem is often regarded as one of the central problems of modern logic. Immense difficulties have arisen which seem to prevent the attainment of satisfactory consistency proofs for languages of sufficient power. Perhaps such proofs will turn out to be impossible. But surely a consistent language-system, whether *provably* so or not, is preferable to one which is known to be inconsistent. Precisely why an inconsistency, in the very basic sense intended here, is so abhorrent to the human intellect is not easy to explain, and perhaps requires a more careful analysis than has heretofore been given it. (See again (V,E).) At any event, it is primarily by means of the study of formalized languages that we can approach the consistency problem at all, for only language-systems are (in the perhaps fundamental sense of the word) consistent or not.[1]

A second important and widely studied meta-theoretic property of formalized language-systems is that of *completeness*. That a system

[1] Of course we often speak of a *consistent* argument, of *consistent* premisses, of *consistent* behavior, etc. But these uses of the word are derivative.

is complete, in an appropriate sense of the word, means roughly that every sentence of that language which is true in a specified interpretation is also provable from the axioms. The axioms of a language-system which is complete in this sense fully characterize the truths of its subject-matter. Another way of saying that a language-system is complete is to say that for every sentence of the language either it or its negative is provable. Within an appropriate meta-theoretic formalism and for an appropriate object-language, these two characterizations of completeness are equivalent. (See (V,F).) Although completeness is a desirable property for a formalism to possess, it is in fact often found lacking, especially in language-systems of great deductive power. Nonetheless it remains a kind of ideal, even though most important scientific and philosophical systems fall short of it. (See (V,H–I).)

Once the primitives or undefined signs of a language have been determined, the definitions introduce new expressions. The definitions must clearly be given in a certain order to prevent circularity. Strictly speaking, defined signs can always be eliminated and the order in which two defined signs are eliminated should be indifferent. Some definitions are such as to introduce abbreviations for expressions already containing defined expressions. In this way a kind of order is established among the definitions, although it is wholly extraneous to the development of the system in primitive terms. But when we consider the interpretation of the system, the order in which we set up the definitions creates a corresponding order in our ideas. We proceed from certain primitive expressions, the interpretation of which is fixed, to longer sequences of primitive expressions (which are abbreviated by the definitions), the interpretation of which is wholly determined by that of the primitives.

Also by the systematic listing of definitions we are better able to compare the consequences of alternative definitions and hence to weigh their relative advantages. Much of systematic thought is concerned with definition. From the strict point of view definitions are mere abbreviations, as we have noted, but, as Whitehead points out, ". . . if we abandon the strictly logical point of view, the definitions . . . are at once seen to be the most important part of the subject. The act . . . [of giving a definition] . . . is in fact the act of choosing the various complex ideas which are to be the special object of study. The whole subject depends upon such a choice."[1] It is only within the framework of a formalized language that definitions in the strict sense can be given, and hence only within such a framework that the

[1] A. N. Whitehead, *The Axioms of Projective Geometry* (Cambridge University Press, Cambridge: 1906), p. 2.

relative advantages or disadvantages of alternative definitions can be carefully determined.

The explicit statement of axioms and rules, and the systematic proving of theorems by means of them, establishes an order and consecutiveness to our assertions or theorems just as the listing of definitions does with respect to our ideas. These kinds of order are often an immense aid in learning and thus useful pedagogically. Students, at whatever level of maturity, once they are familiar with the basic pattern of formalization, find the systematic presentation of a subject-matter a great aid in learning. Once the basic pattern of formalization is mastered, students often become dissatisfied with the amorphous presentation of a subject-matter that is more common. Nor is it at all difficult to master this pattern. The logical notation, as Woodger has put it, "has nothing specially erudite or exotic about it. It enables us to express briefly and precisely ideas (for the most part perfectly familiar) which ordinary language expresses lengthily and ambiguously. Its range of application is extremely wide. . . . Its unfamiliarity soon evaporates with use, and with practice its handling becomes rapid and easy. It is becoming recognized as an indispensable tool in any exact investigation. From the point of view of generality, the essentials of *Principia Mathematica* have as good if not a better claim to be 'taught in schools' as the traditional algebra and geometry."[1]

Although the natural languages suffice for many of the purposes of thought and communication, often much greater precision, particularly in philosophical or scientific discourse, is required. Woodger has made this point forcefully when he writes that a natural language "is not only used for purposes of communication in the scientific sense. It is also used for the writing of poetry, for religious devotion, for political controversy, and for persuading people to buy some of the products of industrial activity which they would not otherwise want. But these pursuits make demands upon language which are very different from those made by science. The requirements of science [and philosophy] prove on investigation to be quite surprisingly meagre and the excessive riches of a natural language like English are a source of embarrassment. They tempt us to employ linguistic devices borrowed from extra-scientific [or extra-philosophical] usages which can have unfortunate consequences. Metaphors, e.g., with which some branches of biology abound, are often suggestive and may be harmless enough if they are recognized for what they are. But at best they are makeshifts and substitutes for

[1] J. H. Woodger, *The Axiomatic Method in Biology* (Cambridge University Press, Cambridge: 1937), p. ix.

genuine biological statements, and the fact that recourse is had to them is surely a sign of scientific immaturity. Science demands great linguistic austerity and discipline, and the canons of good style in scientific writing are different from those of good writing in other kinds of literature."[1]

The formalization of a language-system expressing a given domain of scientific or philosophical theory enables us to bring out clearly the meagre linguistic resources needed and thus to separate out the essential linguistic ingredients from the non-essential ones. Among the essential ones some are found to be common to all systems, others only to one or more particular systems. The common essential ingredients constitute the *logical* vocabulary, and the others, the non-logical or *descriptive* vocabulary. (See Chapter II.) By formalization we gain an explicit enumeration of the logical and non-logical vocabulary needed. This enumeration, together with the definition of 'formula', the stipulation of the axioms, rules, etc., constitutes a precise characterization of the *logical or linguistic structure* of the discipline in question.

Also formalized languages exhibit advantages as media of communication in any well-developed or exact discipline. They are explicitly designed to accommodate semantical rules and the definition of new terms. Communication within such languages is therefore both easier and more efficient than within less exact media.

In urging at such length the importance of formalization for philosophical analysis and the methodology of the sciences, it is not claimed that anything fundamentally new has been said. The view put forward is in essential respects an ancient one, and has frequently been espoused by some of the great figures in philosophical history. It appears to have originated with Plato. Indeed, the method of formalization has been compared with Plato's *dialectic* by A. E. Taylor. "We may say that what the *Republic* calls 'dialectic' is, in principle," he tells us, "simply the rigorous and unremitting task of steady scrutiny of the indefinables and indemonstrables of the sciences, and that in particular, . . . [Plato's] . . . ideal, so far as the sciences with which he is directly concerned goes, is just that reduction of mathematics to rigorous deduction from expressly formulated logical premisses by exactly specified logical methods of which the work of Peano, Frege, Whitehead, and Russell has given us a magnificent example."[2]

The essentials of the methods of formalization and interpretation

[1] *Biology and Language*, pp. 7–8.

[2] A. E. Taylor, *Plato the Man and His Work* (Dial Press, New York: 1936 (first published 1922)), p. 293.

are deeply ingrained in the western mind and perhaps constitute the ideal prototype of some aspects of what we call *rational* thought. Many philosophers and methodologists use the essential features of these methods without perhaps being consciously aware of it. But too much cannot be known of methods constantly employed, and greater awareness of the details of formalization and interpretation would help to raise the standards of philosophical and methodological discussion.

D. Mathematical, Scientific, and Philosophical Languages. The technic of formalization has no doubt had its greatest success to date in application to mathematics. Two kinds of very powerful formalized languages of especial importance for mathematics have been put forward and studied with meticulous care, those based on the *Zermelo set theory* and those based upon the *Russell theory of types*. (See Chapter VI. There are to be sure other derivative kinds of set theory also, but these seem to combine in various ways features to be found in one or the other of these two basic kinds.) It could scarcely be claimed, however, that these languages have as yet proved to be of interest for the *working* mathematician. No doubt the formulation of such languages is an important chapter in the history of mathematical rigor. But rigor is a minor topic in the vast structure of modern mathematics. The most important mathematical discoveries have no doubt always been of a highly intuitive kind, and it is debatable whether the use of formalized languages will prove to be a genuine aid to mathematical *discovery*.

But formalized languages are of interest in research *about* mathematics or various domains of it *as a whole*. Here a new collection of problems has arisen and the study of them has developed into a new science, that of *meta-mathematics*. This is a new domain of twentieth-century mathematics, along with topology, modern algebra, or mathematical statistics. Meta-mathematics presupposes and utilizes the technic of formalization, but only as a preliminary. Problems still remain concerning how best to formalize a given mathematical theory for given meta-theoretic purposes. But formalization *qua* formalization is perhaps not of importance for mathematics except as a prelude to meta-theory. The subject-matter of mathematical languages is so clear-cut and precise and the mathematician's insight into it so vivid and penetrating, that the formalization of a domain of mathematics is often merely a mechanical or routine procedure.

When we turn to a less exact subject-matter, we have a very different situation. The concepts and notions within even the

well-developed physical sciences, e.g., are rarely as precise as in mathematics, the definitions are vague and often ill-formulated, or at best the terms are only partially defined. The nature of the underlying logic is rarely if ever specified, so that whether such and such logical or other principles may be used in a given context, or indeed are used, is often not known. In such areas the method of formalization has much to provide.

Whether the precise technics of formalization and interpretation are useful as aids to *discovery* in empirical science is controversial. Some claim that, because scientific discovery of either facts or of explanatory laws is primarily an intuitive affair, or a matter of good luck, or of trial and error, the refined tools of modern logic are of little help. Others claim that such tools are an aid to discovery of both facts and hypotheses, by drawing attention, e.g., to unexplored areas for observation and by suggesting forms of statement more general than or alternative to accepted hypotheses.[1] If some empirical scientists find such tools useful, this is all to the good. If the methods of modern logic were more widely known by scientists, we should perhaps have a more conclusive argument. And, indeed, it may well turn out that this usefulness has been somewhat exaggerated.

There can be little doubt, however, that the method of formalization is of utility in helping to clarify fundamental issues in the *methodology* of the sciences as distinguished from their actual practice. This has already been suggested above. Contemporary studies in the methodology of science are primarily concerned with the syntax and semantics of the language of science. Already some important studies in formalizing the language of the sciences for methodological purposes and in explicating fundamental concepts common to the sciences have been made. It suffices to mention by way of examples some of the work on the language-structure of Newtonian mechanics, on the theory of relativity, on the foundations of inductive logic and probability, on the theory of measurement, on the behavioral theory of rote learning, and on the explication of some basic concepts in various branches of biology.[2] This is not the occasion to examine

[1] See Woodger's essay, "Mathematical Logic as a Tool of Analysis".

[2] See, e.g., J. C. C. McKinsey, J. Sugar, and P. Suppes, "Axiomatic Foundations of Classical Particle Mechanics", *Journal of Rational Mechanics and Analysis*, 2 (1953); H. Reichenbach, *Axiomatik der Relativistischen Raum-Zeit-Lehre* (Braunschweig: 1924); R. Carnap, *Logical Foundations of Probability* (University of Chicago Press, Chicago: 1950); Clark Hull, F. B. Fitch, *et al.*, *Mathematico-Deductive Theory of Rote Learning* (Yale University Press, New Haven: 1940); and J. H. Woodger, *The Axiomatic Method in Biology* and *Biology and Language*. See also Carl G. Hempel, *Fundamentals of Concept*

any of these in detail. The only point we wish to make here is that formalization *qua* formalization is of fundamental importance in the methodology of the sciences, if not in their actual practice.

Formalization *qua* formalization also has much to contribute to the procedures of philosophical analysis, as has already been suggested. Philosophical analyses are concerned, either directly or indirectly, with explicating or clarifying or interrelating notions or concepts or principles met with in the various sciences, or in discourse concerning experience, literature, art, religion, history, and so on. To explicate or analyze carefully a concept or principle is to incorporate a precise correlate of it within a well-knit theory. (See § E below.) Whenever we speak of a philosophical *theory* or *system*, it is to a formalized language, so it would seem, that we are referring either directly or indirectly, whether it is explicitly formulated or not.

Of course much preliminary spade-work must often be done before an adequate formalization of the results of a philosophical analysis can be made. Often only an approximate formalization is attained, or at best a formalization of only a narrow part of the intended analysis. Here only a *partial* formalization may be achieved.

Let us distinguish between a *fully* and a *partially formalized language* as follows. In §§ A–B seven items constituting a fully formalized language were listed. Of these only four are required for a partially formalized language. If we require only

(i) the primitive vocabulary,
(ii) a definition of 'formula' (and perhaps of 'term'),
(vi) some definitions or abbreviations,

and

(vii) semantical rules,

the result is clearly a *language* in the sense of being an instrument in which communication is possible. But it is only partially formalized in the absence of axioms, rules of inference, and theorems. The character of the underlying logic may or may not be specified. (See Chapter II.) If it is, the *logical* axioms and rules, and therewith theorems, are determined. But a fully formalized language-system must contain a specification of the *non-logical* axioms and rules as well.

Formation in Empirical Science (*International Encyclopedia of Unified Science*, vol. II, no. 7, University of Chicago Press, Chicago: 1952), and R. Carnap, *Foundations of Logic and Mathematics*, *ibid.* (vol. I, no. 3: 1939).

When we speak of the importance for philosophical analysis of formalized languages, we include also partially formalized languages. At the same time we must recognize that partially formalized languages are mere stepping-stones to fully formalized languages, and that one's objective as a philosophical theoretician is only partly achieved if the non-logical axioms and rules have not been fully specified. The axioms and definitions are intimately intertwined. A slight change in one often occasions a fundamental change in the other. It is therefore highly desirable to give the rules and axioms, both logical and non-logical, whenever possible. But owing to the difficulties of providing such, we must often rest content with only partial formalizations.

Partially formalized languages are of especial interest in areas, which have not been intensively developed, or which are concerned, to speak loosely, with a vague or imprecise subject-matter. Formalized languages dealing with a clear-cut and precise subject-matter, as is mathematics, are (as we have seen) of perhaps little interest for the working scientist. But where the subject-matter lacks the clear-cut simplicity of mathematical subject-matter, in such areas formalized languages may be of real help to the working theoretician. The use of a clear-cut vocabulary may help in classifying and ordering the subject-matter dealt with. Perhaps arbitrary decisions will have to be made concerning the extension or degree of a term here or there. Perhaps it will be very difficult to characterize exactly the forms of the sentences allowed. And so on. If we achieve only a partially formalized language as the net result of our analysis, we still may have taken an important step forward.

As a few examples of partially or fully formalized language-systems of outstanding philosophical interest, we may mention the meta-languages (of Tarski and Carnap) containing a semantical concept of truth, the systems explicating the notions of necessity and analyticity or logical truth, those concerned with the notion of degree of confirmation or of evidential support, and those concerned with the analysis of qualities and of their relations to experience and to physical objects.[1] Also we may cite older languages, such as that of *Principia Mathematica* or of the Zermelo set theory, which provide

[1] See, e.g., A. Tarksi, *Der Wahrheitsbegriff in den Formalisierten Sprachen, Studia Philosophica*, I (1936), pp. 261–405; R. Carnap, *Meaning and Necessity* (University of Chicago Press, Chicago: 1947) and "Modalities and Quantification", *The Journal of Symbolic Logic*, II (1946), pp. 33–64; F. B. Fitch, *Symbolic Logic An Introduction* (The Ronald Press, New York: 1952), pp. 64–80 and 164–6; R. Carnap, *Logical Foundations of Probability*, and *Der Logische Aufbau der Welt* (Weltkreis-Verlag, Berlin-Schlachtensee: 1928); and N. Goodman, *The Structure of Appearance* (Harvard University Press, Cambridge: 1951).

theories concerning such philosophically important notions as number, continuity, infinity, and so on.

E. Explication and Rational Reconstruction. Many important philosophical language-systems have been formulated by philosophers associated with logical empiricism. Logical empiricists often speak of their work as providing an *explication* or a *rational reconstruction* of a concept or principle or of a given area of knowledge. "The task of explication," Carnap says, "consists in transforming a given more or less inexact concept into an exact one or, rather, in replacing the first by the second." The inexact concept "may belong to everyday language or to a previous stage in the development of scientific language". On the other hand, the new, more exact concept "must be given by explicit rules for its use, for example, by a definition which incorporates it into a well-constructed system of scientific, either logico-mathematical or empirical, concepts".[1] Much of the task of systematic philosophy is to provide an explication in this sense of various concepts, i.e., not only a careful logical analysis of the older concept, but also the incorporation of the newer concept within a well-knit systematic structure. This systematic structure is merely a formalized language-system, without which genuine explication cannot take place.

One should, strictly speaking, be careful to distinguish the *logical* from the *empiricist* facets of logical empiricism, these being in a certain sense independent of each other. One can no doubt provide an explication of certain concepts in non-empiricist terms. The semantical truth-concept, e.g., does not seem to depend upon the thesis of empiricism; nor does it in any way contradict that thesis. The point is that to provide an explication in Carnap's sense is to use the procedures of logic or formalization fundamentally. But whether the explication is of an empiricist kind depends upon further considerations. Strictly speaking, explication is neutral as between empiricism and anti-empiricism. Certain concepts can perhaps be adequately explicated only in terms meeting empiricist demands, whereas others may in no way depend upon those demands.

The method of explication, involving formalization including interpretation, is no doubt of much wider potential applicability than is generally recognized and is by no means the exclusive possession of any one philosophical school. It would perhaps be generally agreed that explication in this neutral sense is one of the important tasks of philosophy. That this is the case has in effect been suggested in the preceding section. If so, formalization and interpretation may

[1] R. Carnap, *Logical Foundations of Probability*, p. 3.

prove to be of significance for metaphysics, ethics, aesthetics, theology, philosophy of history,[1] etc., whether of empiricist or anti-empiricist kinds. These methods, like logic itself, seem to be neutral with respect to opposing philosophical issues. They are concerned with matters which must be dealt with satisfactorily prior to branching off into specific philosophical or methodological constructions.

F. Paradoxes, Natural Language, and Approximation. It is sometimes claimed that a natural language such as English is inconsistent because of the presence of paradoxes or antinomies. In recent years two kinds of paradoxes have emerged, the *logical* ones and the so-called *semantical* or *epistemological* ones.[2] The logical paradoxes include, e.g., those of Russell and Burali-Forti. (See (VI,C).) Natural language is inconsistent, so some contend, because of the presence of the Russell and other paradoxes. But to establish this definitively would be a task of immense difficulty. For one thing, it would presuppose an at least partial formalization of the natural language concerned.

Some think that it can be claimed more forcefully that natural language contains some at least of the semantical paradoxes. The semantical paradoxes differ from the logical ones in involving essentially in one way or another the words 'designate', or 'name', or 'is true', or the like. E.g., a form of the so-called *Epimenides paradox* (attributed to Eubulides) runs somewhat roughly as follows. Suppose one says,

(1) 'This sentence is not true',

where 'this sentence' refers to the very sentence (1). (1) is itself presumed to be a sentence and hence is presumed to be either true or false. Suppose it is true. Then what it says in fact obtains, and hence it is not true. On the other hand, suppose (1) to be false. Then what it says does not obtain, and hence it is true. The sentence (1) is thus true if and only if it is not true, which is contradictory. (1) is presumed to be a legitimate sentence of natural English, and the various laws used (tacitly) here in the derivation of the contradiction are presumed to be acceptable laws within natural English. There can be little doubt, if these presumptions are correct and if this argument does full justice to the matter, that natural English contains this semantical antinomy and hence is contradictory.

[1] See, e.g., P. Schrecker's *Work and History, An Essay in the Structure of Civilization* (Princeton University Press, Princeton: 1948), which may very profitably be studied in this connection.

[2] See, e.g., F. P. Ramsey, *The Foundations of Mathematics* (Kegan Paul, London: 1931).

It is not essential here to take sides one way or another concerning the consistency of natural language. One cannot settle this issue definitively perhaps until much more is known about the precise syntactical and semantical structure involved, as has already been suggested. The analysis of denotation and meaning in natural language presents problems of great difficulty which cannot be solved by casual methods. The precise formulation of these problems, let alone the solution of them, requires presumably the development of careful empirical methods of analysis. No doubt empirical linguistics will have much to contribute to such development,[1] and so perhaps will social and psychological studies concerning linguistic behavior and reactions.

Meanwhile an important approach to the problems of denotation and meaning in natural language, and one frequently neglected by its students, is to be found in the study of formalized languages of greater and greater power of expressiveness. This approach may be called the *approximative* or *abstractive* approach to natural language.

In any exact investigation, e.g., in developing a theory about a given subject-matter, one abstracts, so to speak, from its total complexity. One singles out certain features for explicit study. In singling these out one must neglect other features. But in this neglect we achieve a narrowing of the subject-matter which renders it amenable to exact characterization. Such abstraction is perhaps an essential concomitant of systematic thought.

The importance of abstraction in the study of certain problems in the methodology of science has been recently pointed out by Carnap. ". . . [B]y the abstraction which we carry out in order to construct our system . . .," he says, "we disregard certain features; . . . Some philosophers say that, in consequence of the abstraction leading to logic or, in a similar way, to quantitative physics, certain features of reality (for instance, the 'genuine qualities' or 'qualia') remain forever outside our grasp. I do not agree with this view," he goes on. ". . . This may become clearer by the following analogy. Suppose a circular area is given, and we want to cover some of it with quadrangles which we draw within the circle and which do not overlap. This can be done in many different ways; but, whichever way we do it and however far we go with the (finite) procedure, we shall never succeed in covering the whole circular area. However, it is not true

[1] See, e.g., Z. Harris, *Methods in Structural Linguistics* (University of Chicago Press, Chicago: 1951). Also C. Morris, *Foundations of the Theory of Signs* (*International Encyclopedia of Unified Science*, vol. I, no. 2, University of Chicago Press, Chicago: 1938), and *Signs, Language, and Behavior* (Prentice-Hall, New York: 1946).

that—in analogy to the philosophical view mentioned—there is any point in the area which cannot be covered. On the contrary, for every point and even for every finite number of points there is a finite set of quadrangles covering all of them. The situation with abstraction is analogous. In any construction of a system of logic or, in other words, of a language-system with exact [syntactical and semantical] rules, something is sacrificed, is not grasped, because of the abstraction or schematization involved. However, it is not true that there is anything that cannot be grasped by a language-system and hence escapes logic. For any single fact in the world, a language-system can be constructed which is capable of representing that fact while others are not covered. . . ."[1]

"Some scientists and philosophers", Carnap notes, "feel a strong disinclination against all abstractions or schematizations. They demand that any methodological or even logical analysis of science should never lose sight of the actual behavior of scientists both in the laboratory and at the desk. They warn against neglecting any of the factors which a good scientist takes into consideration in inventing and testing his hypotheses . . . I think that this view contains a correct and important idea. Whenever we value an abstraction, we certainly ought to be fully aware of what we are doing and not to forget that we leave aside certain features of the real processes and that these features from which we abstract at the moment must not be entirely overlooked but must be given their rightful place at some point in the full investigation of science. On the other hand, if some authors exaggerate this valid requirement into a wholesale rejection of all schematizations and abstractions, . . ., then they deprive science of some of its most fruitful methods.

"The history of science is full of examples for the usefulness and immense fertility of abstractions. One of the most outstanding examples is geometry. It was created by an act of abstraction: attention was directed toward the spatial properties and relations of bodies, while all other properties, color, substance, weight, etc., were disregarded. Then another bold step was taken, leading away from the world of concrete things with their directly observable properties to a schema consisting of constructs: geometry was transformed into a theory of certain spatial configurations whose properties are completely and exactly determined. This geometry no longer deals with wooden or iron balls but with spheres, perfect spheres of which the balls are only more or less rough approximations. It deals with infinite straight lines, of which at best some finite segments are approximately represented by certain threads and edges of

[1] R. Carnap, *Logical Foundations of Probability*, p. 210.

bodies. . . . Today it is clear that the magnificent development of geometry through its history of more than two thousand years would have been impossible without those abstractions and that the development of physics would have been impossible without that of geometry. Thus the end result is that, not only from the point of view of the mathematician but also from that of the physicist, the abstractions in geometry are immensely useful and even practically indispensable. Although the aim remains the investigation not of the abstract configurations but of the observable spatial properties of concrete things, nevertheless it turns out that abstract geometry supplies the most efficient method for this investigation, much more efficient than any method dealing directly with observable spatial properties. Numerous other methods of abstraction or schematization have proved fruitful in physics. This shows that, if we want to obtain knowledge of the things and events of our environment as a help for our decisions in practical life, then the roundabout way which leads first from these things to an abstract schema may in the long run be better than the direct way which stays close to the things and their observable properties."[1]

Carnap's forceful remarks have been quoted at such length because they provide in outline an argument in behalf of the approximative approach toward natural language. This approach presumes that one can best approximate the expressive power of natural language by formalized languages *constructed for explicit scientific, methodological, or philosophical purposes*. This approach has been called both 'abstractive' and 'approximative'. We may start our investigation with natural language and whittle it down, so to speak, for specific methodological purposes, or we may start with very narrow formalized languages and construct systems of greater and greater complexity. The end result is essentially the same in both methods. The resulting formalized language may contain only a small portion of the expressive power of a natural language. But every theoretical part of natural language can be represented in this way by some formalized system or other. Every formalized language is thus presumably contained in natural language. An exact analysis of denotation and truth, of the kind to be presented in this book for formalized languages, should therefore be contained in any adequate analysis of allied notions for natural language.

To summarize. In § C above seventeen reasons were given as to why formalized languages are thought to be important for philosophical analysis and methodology. (1) A clarity and precision is gained in the actual procedures of formalization which are perhaps

[1] R. Carnap, *ibid.*, pp. 215–17.

equally attainable in no other way. The use of formalized languages provides (2) an aid to intuition, (3) a corrective to faulty thinking, and (4) a systematic method of criticizing obscure ideas. (5) A formalized language aids in making sure that all the assumptions in a given context are explicitly laid down. (6) Formalized languages enable us to elicit in a systematic way the consequences of our assumptions. By reference to formalized language-systems we can make the important distinctions between (7) meta-language and object-language and (8) syntax and semantics. (9) An explicit interpretation of the primitive symbols of a formalized language can be laid down. (10) Formalized languages provide a basis for the study of metatheoretic properties, for the comparison of theories, for the reduction of one theory to another, etc. Only formalized languages can, in the most proper sense, be said to be (11) consistent or (12) complete. (13) The use of clear-cut definitions, of carefully laid down axioms, and explicitly proved theorems creates a corresponding order in our ideas and assertions, which is (14) pedagogically useful as an aid in learning. (15) The use of formalized languages enables us to characterize explicitly the logical structure of a science or other well-developed discipline. (16) They provide precise instruments of communication and (17) constitute a kind of ideal to which systematic thought aspires. Further reasons emerge when we consider (18) the role of explication or rational reconstruction, (19) the possible presence of paradoxes in natural language, (20) the necessity for abstraction in all exact discourse, and (21) the possibility of approximating for specific scientific, methodological, or philosophical purposes the expressive power of natural language by means of formalized languages. These twenty-one arguments are not all independent of each other, nor are they by any means exhaustive. Also many of them are somewhat rough and in need of a more thorough elaboration than can be given here.[1]

G. Minimal Core of Semantical Theory. We may now state the main task of this book, namely, to provide methods of giving the

[1] To prevent misunderstanding, it should be noted that, in connection with the arguments concerning the importance of formalization for philosophy, we have been talking here mainly of philosophical *systems*. Philosophical *arguments* may sometimes be couched in philosophical systems, but not always. Arguments are usually, although not always, at the meta-linguistic level and frequently utilize sentences other than declarative ones. Such arguments cannot be formalized in the sense discussed here on the basis of present knowledge. Cf., e.g., G. Ryle, *Philosophical Arguments* (Inaugural Lecture, Oxford University Press, Oxford: 1945) and *Dilemmas* (Cambridge University Press, Cambridge: 1954), Chapter VII.

interpretation of formalized logistic systems in what are thought to be the simplest and most economical ways possible.

Given a formalized logistic system, there are many methods of laying down an interpretation for it which have been carefully studied in the literature of modern logic. All these methods are fairly elaborate and use fundamentally logical and mathematical devices of great complexity. Also these methods are usually of a somewhat restricted applicability, suited primarily for the semantical analysis of some one object-language or for object-languages of some one (perhaps narrow) kind.

Here we are concerned primarily with semantical methods of extreme simplicity and of very broad generality. Several alternative methods will be developed, and some of them will be preferable to others in certain respects. When compared with other formulations of semantics, the methods here will be seen to be especially simple and straightforward. No complicated assumptions by way of axioms or rules are required. The languages in which the semantics is couched are of a very simple kind, and the only mathematical notions used are (roughly) those of elementary arithmetic. The methods here are also of very wide generality, applicable in fact to all first-order object-languages. Most of the important formalized languages which have been put forward are either directly of first order or can readily be reformulated as such. The methods developed here are therefore applicable to almost all formalized languages.

We shall formulate in this book a kind of *minimal* core of denotational semantical theory. For this definitions as to what is meant by 'semantics' and by 'minimal' will be given, and it may be that complete agreement concerning the adequacy of these definitions will not be forthcoming. Convincing reasons will be put forward in favor of them, however, and it will be proved that certain of the semantical theories formulated are minimal in the sense defined. The first theories to be presented are clearly not minimal, being based essentially on traditional methods.

The various semantical theories to be formulated have the important property of being consistent relative to the assumptions of the underlying syntax and of the object-language. Thus the semantical methods here are secure in the sense of containing no contradiction, unless the underlying syntax or object-language does. Because of the possible presence of semantical paradoxes in natural languages and the likelihood that these may reappear in any language containing semantical ingredients, the relative consistency of the methods developed is perhaps of especial interest.

This minimal core of semantical theory contains as a part a

restricted kind of syntax. This syntax and semantics together may be thought of as constituting a kind of minimal linguistic doctrine upon which very wide agreement ought to be possible. Other kinds of denotational semantics, utilizing more powerful logical means of deduction and more complicated notions, would not seem to provide a basis for such agreement. For one thing, their relative consistency might be in doubt. And for another, agreement on the powerful assumptions needed might not be forthcoming. But the assumptions used here are so restricted that even philosophers and methodologists of very different kinds can perhaps agree upon them. At least this much linguistic doctrine might therefore be regarded as a necessary prolegomenon for all systematic philosophical or methodological analyses!

H. Summary. The main content of this book may be summarized very briefly as follows.

An important feature of most of the theories to be presented is that denoting expressions are taken in what appears to be essentially the old sense of Hobbes. "Of names," according to him, "some are *Proper* and singular to one onely thing; as *Peter*, *John*, . . .: and some are *Common* to many things; as *Man*, *Horse*, *Tree*: every one of which though but one name, is nevertheless the name of divers particular things; . . ."[1] According to this, names or denoting expressions may be either proper or common, i.e., they may denote or stand for just one specific object, or they may stand severally for any one of a multitude of objects. Hobbes' usage frequently occurs in British logical writing; it is to be found in the works of Mill, J. N. Keynes, Jevons, and others. This usage is somewhat wider than is customary in philosophical, logical, and semantical writing today. Common names or denoting expressions are now almost universally called 'predicates', and 'name' is now used exclusively in the sense of 'proper name'. This latter terminology has become customary, partly no doubt because it is that of Russell and Carnap.

But there are other reasons also why 'name' should have come to be used as a synonym for 'proper name'. These seem to revolve around a basic supposition which has dominated semantical thought during the last two or three centuries. This basic supposition has been called by Carnap the *Principle of Univocality*. (Carnap does not necessarily advocate this principle himself; but he seems to have been the first to formulate it clearly and to realize the important role it has played.) The Principle of Univocality states roughly that every denoting expression in a given language denotes exactly one entity.

[1] Th. Hobbes, *Leviathan*, Cambridge edition, 1904, p. 15.

Hence to be a name is to be a proper name, whether the name of an individual, class, property, function, truth-value, or proposition. Names which denote more than one object of any kind are not properly names at all.

The Principle of Univocality has played an important role historically. A leading feature of many of the semantical theories to be developed here, however, is the rejection of this principle, so that denoting expressions or phrases are allowed to stand severally for more than one object. But of course proper names, which denote just one object, may also be admitted if desired. Proper names and expressions denoting more than one object have many semantical properties in common, and are perhaps more closely allied conceptually here than within a semantics employing a univocality principle. But we shall use 'name' throughout exclusively in the sense of 'proper name'. This usage is now so well established that little would be gained by extending it to include common names.

The underlying logic to be used throughout this book is the kind ordinarily called *first-order logic*, which comprises only the theories of truth-functions, quantifiers, and perhaps identity, the most elementary domains of modern (classical, two-valued) logic. In order to make the book wholly self-contained, an exposition of first-order logic of the kind needed will be given in the next chapter. The formulation there is essentially that of *Principia Mathematica* in the form due to Church. All the language-systems to be dealt with in any detail, either as object-languages or as syntactical or semantical meta-languages, are, in the terminology of Church, *simple, applied,* functional calculi of first order with or without identity.

Semantics presupposes syntax and contains it as a part. As a prelude to a first-order semantics, we shall therefore need a first-order syntax. This will occupy Chapter III. The kind of formulation given is due to Chwistek, Tarski, and Quine.

The discussion of semantics proper starts with Chapter IV. The first semantical theory to be presented is based upon a relation called *multiple denotation*. Multiple denotation is distinguished from singular denotation just as, in a similar way, common denoting expressions are distinguished from proper names. A common denoting expression bears the relation of multiple denotation to (or denotes severally) the objects to which it applies. In terms of multiple denotation, the other basic notions and relations of semantics are definable, including the semantical truth-concept. In Chapter IV the definitions of various semantical notions are given along with the Rules of Denotation, the semantical rules stipulating the properties of multiple denotation.

In Chapter V several alternative definitions of 'truth' are given and their *adequacy* (in essentially the sense of Tarski) proved. Also the important uses of the semantical truth-concept, in connection with consistency and decidability, are brought out. A proof of the relative consistency of the semantical meta-language based upon multiple denotation is given, on the supposition that the axioms of syntax together with those of the object-language are consistent. Finally there is brief discussion of the semantical antinomies and of how and why the semantical meta-language at hand cannot consistently contain its own truth-concept.

Often formulations of denotational semantics are such that they do not directly supply a truth-definition for such powerful language-systems as those based upon the simplified theory of types or upon the Zermelo set theory. In terms of multiple denotation, a semantics for such powerful languages can be given in the same uniform way as for simpler systems. Chapter VI is devoted to showing this in detail. An informal, preliminary discussion of these important languages is given for those not familiar with them.

In Chapter VII three alternative kinds of semantical meta-languages are considered, one based upon a very simple relation of *satisfaction*, one on a simple form of singular denotation or *designation*, and one on *determination*. Designation and determination are especially appropriate primitives for the semantics of object-languages based upon the theory of types.

The various semantical meta-languages of Chapters IV–VII are closely patterned after meta-languages studied by Tarski and Carnap.

All the semantical meta-languages which have been studied heretofore, including those developed in Chapters IV–VII, have the common feature of containing a translation *in toto* of the object-language to which they are applied. Such meta-languages may reasonably be called *translational* meta-languages. In Chapter VIII the framework of a *non-translational* semantical meta-language is sketched. The meta-language is non-translational in the sense that it does not contain a strict translation of the object-language. Non-translational semantics affords a radical departure in some respects from the semantics of the preceding chapters and it is thought that considerable philosophical interest attaches to it. The only semantical primitive, in one formulation, is a relation of *comprehension*, according to which one predicate constant is said to comprehend another. E.g., in an appropriate formalism, 'animal' may be said to comprehend 'man', or 'lover of peace', 'Quaker'. Comprehension relates predicate constants with predicate constants whereas multiple denotation and the other semantical relations mentioned above

relate predicate constants with non-linguistic objects. Taking comprehension as the sole semantical primitive, one can build up a kind of extended *Boolean algebra* in which the fundamental elements are predicate constants. The semantical truth-concept is readily definable within such a theory. Several alternative formulations of non-translational semantics are given, which lead to a proof that it constitutes a *minimal* semantics in the appropriate sense.

In Chapter IX, several theorems of non-translational semantics are proved. It is shown that on certain hypotheses all sentences of the appropriate object-language which are theorems are also true. This is one of the most fertile theorems of semantics and leads to the proof of the consistency of the object-language. Within non-translational semantics one can prove on suitable hypotheses all the important theorems which are provable in ordinary translational semantics. A proof of the consistency of the axioms of non-translational semantics is also given, relative to the consistency of the axioms of the underlying syntax.

The formulation of the semantical meta-languages considered is simplified somewhat if the object-languages to which they apply do not contain primitively individual constants (proper names) or functional constants (functors). In Chapter X the slight complications which the introduction of individual or functional constants as primitives requires, are considered first in the semantics based upon multiple denotation and then in that based upon comprehension. It is shown also that a non-translational semantical meta-language can be formulated in such a way as to contain consistently its own semantical truth-concept.

One of the merits of the various (translational and non-translational) semantical meta-languages formulated is that they give rise, with only minor changes, to semantical theories in which the expressions of the object-language may be regarded as *inscriptions* or *sign-events*, not *shapes* or *sign-designs*. Semantics heretofore has been based almost exclusively upon sign-designs. An inscriptional theory is formulated in Chapter XI, a form of multiple denotation being taken as a primitive. This semantics is therefore of the translational kind.

In Chapter XII a simplification of inscriptional semantics is given, in which a relation of comprehension is taken as sole semantical primitive. This theory is therefore non-translational. A non-translational semantics based upon inscriptions is thought to give a solution to the problem of gaining a *nominalist* theory of truth, in the very restricted sense of Goodman and Quine.

Chapter XIII is devoted to some general philosophical observations

concerning the various semantical meta-languages formulated. Several reasons for preferring first-order languages, either as object-languages or as meta-languages, are given. Also the reasons for interest in non-translational semantics are enumerated in some detail. The general philosophical view concerning languages which emerges is called *first-order constructivism*. According to this view, first-order languages having denumerable domains of objects as values for variables, are of especial philosophical interest.

In short, this book contains (1) a rather full exposition of some semantical theories closely akin to those first studied by Tarski and Carnap (Chapters IV–VII and parts of X, Chapters II and III being preparatory); (2) the formulation of a minimal semantical theory of a non-translational kind, which affords a radical departure in some respects from the semantical theories developed heretofore (and under (1)) and over which it has some important advantages (Chapters VIII–IX and parts of X); (3) the formulation of two kinds of semantical theories, one translational and one non-translational, in which inscriptions or sign-events are taken as values for expressional variables (Chapters XI–XII); and (4) an argument concerning the importance for philosophical analysis and the methodology of science of denumerable formalized language-systems of first order (Chapter XIII). A denumerabilist philosophy of language is developed hand in hand with the various alternative formulations of semantics.

CHAPTER II

FIRST-ORDER LANGUAGES

THE variables of any first-order language, of the kind to be discussed in this book, are to be regarded as ranging over a well-specified, non-empty domain of *individuals*. The meaning of 'individual' is, for the present, left unspecified so as to be able to give it different meanings in different interpretations of the language. No other kinds of variables are to be admitted, so that first-order languages have a very simple and elementary structure. In particular no property or class variables are admitted, no relational or functional variables, and no propositional variables. There will be no need for variables of these additional kinds. The primitive constants are of two different sorts, logical and non-logical. The non-logical ones comprise individual constants, predicate constants, or functional constants (functors). The individual constants may be thought of as standing uniquely for specific individuals. The predicate constants are ordinarily thought of as standing for properties of or relations between or among individuals. A *one-place* predicate constant stands for a property, a *two-place* predicate, for a dyadic relation, etc. A one-place predicate constant is said to be of *degree* one, a two-place predicate constant, of degree two, and so on. The functors stand for functions or operations applying to individuals as arguments and having only individuals as functional values. Here likewise we can speak of a *one-place* functor, a *two-place* functor, and so on. (A one-place functor 'f' stands for a function f which when applied to an *argument* x has the *functional value* fx.) The logical constants admitted must be such as to provide for the *connectives* 'and', 'or', 'not', etc., as well as for the *quantifiers* on individual variables, expressions such as 'for all x' and 'there is at least one x such that'.

Let us turn immediately to a more precise characterization of first-order languages. The *primitive framework* of such languages will be sketched in § A. Some of the basic theorems of the elementary theory of *truth-functions* will be given in § B and the basic theorems of *quantification theory*, in § C. § D will be concerned with introducing a convenient notation for an operation of *abstraction*. The axioms for the theory of *identity* will be given in § E. In § F theories of *singular* and *selective descriptions* will be developed. Two or three theorems

concerning some useful *normal forms* will be stated in § G. The notion of the fundamental *domain of individuals* of a first-order language will be discussed in § H somewhat more fully than above. The distinction between *object-language* and *meta-language* will be made more precise in § I, that between *syntax* and *semantics* in § J, that between *sign-designs* and *sign-events* in § K. Several *examples* of first-order object-languages will be given in § L.

A. Logical Systems of First Order.[1] We first specify *in toto* the primitive vocabulary. This will consist of (1) the logical constants,

'v', '$\sim$', '(', and ')',

(2) the infinite list of variables,

'x', 'x'', 'x''', . . . ,

and (3) a finite list of predicate constants,

'P', 'P$''$', 'P$'''$', . . . , 'P$'''\cdots''$'.

The logical constant 'v' ('or') is to be thought of as standing for the connective (truth-function) of inclusive, logical *disjunction*, as is customary. The constant '$\sim$' ('not') is to stand for *negation*, in the customary meaning in classical, two-valued logic. The logical constants '(' and ')' need not be regarded as standing for any particular kind of entity; they function rather as syncategorematic or contextual expressions which facilitate grouping or punctuation. The variables are all of one kind, ranging over any well-defined, clear-cut, non-empty domain of individuals. (See § H below.) There is to be an infinite number of variables, so that we can always find a new one if needed. The predicate constants 'P', etc., can be of any finite degree. For definiteness let us suppose for the moment that the languages at hand contain just *five* primitive predicate constants, and that 'P' is

[1] The formulation here is in essential respects that of *Principia Mathematica* in the form due to Church. See A. Church, *Introduction to Mathematical Logic*, Part I (Princeton University Press, Princeton: 1944), pp. 33–98. Cf. also D. Hilbert and W. Ackermann, *Grundzüge der Theoretischen Logik* (Springer, Berlin: 1938 (2nd ed.)); D. Hilbert and P. Bernays, *Grundlagen der Mathematik*, vol. 1 (Springer, Berlin: 1934), pp. 87–209; S. C. Kleene, *Introduction to Metamathematics* (Van Nostrand, New York: 1952); J. B. Rosser, *Logic for Mathematicians* (McGraw-Hill, New York: 1953); and W. V. Quine, *Mathematical Logic* (1st ed., Norton, New York: 1940; revised ed., Harvard University Press, Cambridge: 1951). See also I. Copi, *Symbolic Logic* (Macmillan, New York: 1954) and H. Leblanc, *An Introduction to Deductive Logic* (Wiley, New York: 1955).

(*Note added in proof:* A revised and enlarged edition of the work by Church mentioned has recently appeared under the same title (Princeton University Press, Princeton: 1956).)

of first degree, 'P′′' of second degree, and so on. This supposition is quite inessential but will facilitate some later definitions.

One might also wish to admit some further kinds of primitive expressions. In particular, one might wish to have available primitively some individual constants or some functors. Let

$$\text{'}\bar{a}\text{'}, \text{'}\bar{a}''\text{'}, \text{'}\bar{a}'''\text{'}, \ldots,$$

be individual constants, and let

$$\text{'}\bar{f}\text{'}, \text{'}\bar{f}''\text{'}, \text{'}\bar{f}'''\text{'}, \ldots,$$

be functors each of specified degree. An individual constant is to be regarded as naming or designating uniquely one of the individuals of the fundamental domain. The functors are to be regarded as naming certain functions or operations taking individuals exclusively as arguments and having individuals exclusively as functional values. The presence of individual or functional constants is not essential, and if desired they can be introduced by means of definitions in terms of appropriate predicate constants. (See § F below.) Many of the languages we shall be concerned with, however, will contain them, and therefore we include them here.

Within different languages the same (non-logical) primitive predicate, functional, and individual constants may stand for different properties or relations, functions, or individuals respectively. The logical constants, however, do not strictly stand for any objects, and their behavior is essentially the same in all the languages under consideration. (See (VI,F) below.)

The notion of a *term* may be defined recursively, for the language-systems being discussed, as follows.

1. Any variable or individual constant is a term.
2. Any result of writing an n-place functor followed by n terms is a term.

The notion of *formula* may be defined similarly, presupposing that of term.

1. Any result of writing an n-place predicate constant followed by n terms is a formula.
2. If $\boldsymbol{A}$ is a formula, '$\sim\boldsymbol{A}$' is also.
3. If $\boldsymbol{A}$ and $\boldsymbol{B}$ are formulae, then '$(\boldsymbol{A} \vee \boldsymbol{B})$' is also.
4. If $\boldsymbol{A}$ is a formula and $\boldsymbol{x}$ is a variable, then '$(\boldsymbol{x})\ \boldsymbol{A}$' is a formula.

(Further, the formulae and terms are to be understood as just those sequences of symbols explicitly stipulated to be such by these definitions.)

According to the definition,

$$\text{‘}(\mathrm{P}'\bar{\mathrm{a}}x' \vee \sim\mathrm{P}''xx'x'')\text{’},$$

e.g., is a formula. But

$$\text{‘}(())\bar{\mathrm{a}}x''\mathrm{P}\vee\vee\sim\sim\sim\text{’}$$

is not a formula at all, although it is a sequence of primitive expressions of the language. Similarly ‘$\bar{\mathrm{f}}x\bar{\mathrm{a}}$’ is a term if ‘$\bar{\mathrm{f}}$’ is a two-place functor. And so on.

Note that these informal definitions are recursive in the sense roughly described in (I,A). In the definition of ‘formula’, e.g., clause (1) defines this term as applied to the simplest, i.e., the *atomic* formulae. Clauses (2), (3), and (4) then tell us that, given certain formulae, certain other expressions constructed out of them are also formulae. In this way, formulae of all possible finite lengths are defined. Intuitively speaking, a formula can be described as any expression constructed by means of a finite number of applications of negation, disjunction, or universal quantification starting with atomic formulae. But we cannot *define* the notion in this way unless we have previously defined the notion embodied in the phrase ‘finite number of applications’. Using recursive definitions we see that all possible formulae are covered and that such definition gives a kind of implicit analysis of this phrase. (Cf. (III,D) below.)

Hereafter let ‘***A***’, ‘***B***’, and ‘***C***’, possibly with primes or numerical subscripts, be used to stand for any formula. And let ‘***x***’, ‘***y***’, and ‘***z***’, with or without primes or numerical subscripts, be used to stand for any term. These bold-face letters are (syntactical) variables used informally to range over certain kinds of expressions of the language or languages under consideration. Throughout the book bold-face letters will be used in this way.

When a bold-face letter occurs within a context containing one or more of the primitive symbols of the language, as in (2), (3), or (4) of the definition of ‘formula’ just given, the quotation marks are used in a special sense. (2), e.g., is not to be taken literally as stating that the result of writing ‘$\sim$’ followed by ‘***A***’ is a formula, but rather that the result of writing ‘$\sim$’ followed by ***A*** (i.e., by an expression over which ‘***A***’ ranges) is a formula. Similarly (3) states that the result of writing ‘(’ followed by ***A*** followed by ‘v’ followed by ***B*** followed by ‘)’ is a formula. When single quotation marks enclose an expression containing at least one bold-face variable and at least one of the primitive constants or variables of the language, the single quotation marks have the effect of what are sometimes called *quasi-quotes*. An expression ‘(***A*** v ***B***)’ thus is to stand for the expression which one

forms by filling the respective blanks of

'(v)'

by the expressions ***A*** and ***B***. Similarly '(***x***)***A***' is to stand for the expression one forms from

'( ) '

by filling the first blank by a variable ***x*** and the second by the formula ***A***. If an expression contains no bold-face letter at all and is enclosed in single quotation marks, the quotation marks are used in their usual sense. Although we shall employ single quotes ambiguously in what follows, no possible confusion can result. If the symbolic context contained in quotes contains at least one primitive symbol of the language and at least one bold-face letter, the quotes are being used in the sense of quasi-quotation; but if it contains no bold-face letter at all, the quotation marks are being used in their normal sense.[1] (Note that if a bold-face letter is used *solo* in a verbal context and is not enclosed in quotes, that context is within a meta-language; if it is enclosed in quotes, the context is within a meta-meta-language. See § I below.)

Before giving the axioms and rules of the language, it is convenient to introduce some abbreviations or definitions. Hereafter we let

D1. '(***A*** ⊃ ***B***)' abbreviate '(∼***A*** v ***B***)',
D2. '(***A*** . ***B***)' abbreviate '∼(∼***A*** v ∼***B***)',
D3. '(***A*** ≡ ***B***)' abbreviate '((***A*** ⊃ ***B***) . (***B*** ⊃ ***A***))',

and

D4. '(E***x***)' abbreviate '∼(***x***)∼', where ***x*** is a variable.

It is easy to show that the symbols introduced in this way have their accustomed meanings. The '⊃' stands for the connective (truth-function) of the *material conditional*, the '.' for *logical conjunction*, and '≡' for *material equivalence*, all of these notions understood in the sense that is customary in two-valued, truth-functional logic.

As is evident from the definition of 'formula' above the idiom of universal quantification is expressed by '(***x***)', where ***x*** is a variable, appropriately prefixed to a formula. The formula '(x)(Px v ∼Px')', e.g., may be read 'For all individuals x, x has the property P or it is not the case that x' has P'. We note then that *D4* introduces a notation for *existential quantification*. '(Ex) . . . . . ' may be read 'There

[1] The use of bold-face letters here and throughout as syntactical variables is due to Church. See, e.g., his *The Calculi of Lambda-Conversion* (Princeton University Press, Princeton: 1941). The device of quasi-quotation is due to Quine. See, e.g., his "Logic Based on Inclusion and Abstraction", *The Journal of Symbolic Logic*, 2 (1937), pp. 145–52, and *Mathematical Logic*, pp. 33–7.

is at least one x such that . . . . . '. This is an abbreviated reading of 'It is not the case that for all x, it is not the case that . . . . . '. (In place of '. . . . . ' here we may put in any formula of the language.)

Let us use Frege's compound sign '$\vdash$' (of truth-assertion) in talking about the language to stand for 'is a theorem'.[1] '$\vdash \boldsymbol{A}$' is to be read 'The formula $\boldsymbol{A}$ is a theorem'. And let '$\vdash$' absorb the single quotation marks enclosing any symbolic context following it whether containing a bold-face letter or not. Thus '$\vdash (\boldsymbol{A} \vee \boldsymbol{B})$' is to be read ' '$(\boldsymbol{A} \vee \boldsymbol{B})$' is a theorem', and '$\vdash \mathrm{P}x$' is to be read ' '$\mathrm{P}x$' is a theorem'.

Rather than cite axioms or postulates directly we make certain general statements about the language which state that any formula (of the language) of such and such a form is an axiom. All such general statements are *meta-axioms*. An axiom then is any formula of the language stipulated by one of the meta-axioms. Meta-axioms are a special kind of *rule*. (See § B below.)

The first four meta-axioms specify the *Truth-Functional Axioms*.

R1. $\vdash (\boldsymbol{A} \vee \boldsymbol{A}) \supset \boldsymbol{A}$.
R2. $\vdash \boldsymbol{A} \supset (\boldsymbol{A} \vee \boldsymbol{B})$.
R3. $\vdash (\boldsymbol{A} \vee \boldsymbol{B}) \supset (\boldsymbol{B} \vee \boldsymbol{A})$.
R4. $\vdash (\boldsymbol{A} \supset \boldsymbol{B}) \supset ((\boldsymbol{C} \vee \boldsymbol{A}) \supset (\boldsymbol{C} \vee \boldsymbol{B}))$.[2]

(Note that we have omitted here some unnecessary outer parentheses. We shall make such omissions where no ambiguity can result, and also shall frequently supplant parentheses by dots, colons, triple dots, etc., in a more or less obvious fashion.)

Each of these rules is a statement about the language being discussed, not a statement within the language. Each rule or meta-axiom given thus far stipulates an infinity of formulae of the language, specifying them to be axioms. E.g., *R1* stipulates that

$$'(\mathrm{P}x \vee \mathrm{P}x) \supset \mathrm{P}x'$$

is an axiom, and also that

$$'(((\mathrm{P}x \vee \mathrm{P}x) \supset \mathrm{P}x) \vee ((\mathrm{P}x \vee \mathrm{P}x) \supset \mathrm{P}x)) \supset ((\mathrm{P}x \vee \mathrm{P}x) \supset \mathrm{P}x)'$$

is an axiom. As already remarked, a *rule* is a statement of a general kind about the language being discussed, whereas an axiom is always a specific formula within the language.

[1] See G. Frege, *Grundgesetze der Arithmetik, Begriffsschriftlich Abgeleitet* (vol. 1, Jena: 1893).

[2] These rules are due essentially to Bernays. See P. Bernays "Axiomatische Untersuchung des Aussagen-Kalküls der *Principia Mathematica*", *Mathematische Zeitschrift*, 25 (1926), pp. 305–20. *R3* may be dropped if *R4* is altered slightly by interchanging the right-most occurrences of '$\boldsymbol{C}$' and '$\boldsymbol{B}$'. See the papers by E. Götlind and Helena Rasiowa, *Norsk Matematisk Tidsskrift*, 1947, pp. 1–4, and 1949, pp. 1–3.

Before stating the remaining axioms it is necessary to define the notion of a *free variable*. An occurrence of a variable $\boldsymbol{x}$ in a formula $\boldsymbol{A}$ is said to be a *bound* occurrence of $\boldsymbol{x}$ in $\boldsymbol{A}$ if and only if it is in a part of $\boldsymbol{A}$ of the form '$(\boldsymbol{x})\boldsymbol{B}$'. Otherwise that occurrence is a *free* occurrence. Also any occurrence of an individual constant in $\boldsymbol{A}$ is free in $\boldsymbol{A}$. And an occurrence of a term will, in general, be free in $\boldsymbol{A}$ provided every occurrence of a variable in that term is a free occurrence of that variable in $\boldsymbol{A}$. E.g., the occurrence of the term '$\bar{\mathrm{f}}\bar{\mathrm{a}}x$' (where '$\bar{\mathrm{f}}$' is a two-place functor) in '$\mathrm{P}'x\bar{\mathrm{f}}\bar{\mathrm{a}}x$' is free, but the occurrence of this term in '$(x)\mathrm{P}'x\bar{\mathrm{f}}\bar{\mathrm{a}}x$' is not.

The rules stipulating the *Axioms of Quantification* may now be given as follows.

R5. $\vdash (\boldsymbol{x})\boldsymbol{A} \supset \boldsymbol{B}$, where $\boldsymbol{B}$ differs from $\boldsymbol{A}$ only in containing free occurrences of some one variable or term $\boldsymbol{y}$ wherever there are free occurrences of the variable $\boldsymbol{x}$ in $\boldsymbol{A}$.

R6. $\vdash (\boldsymbol{x})(\boldsymbol{A} \vee \boldsymbol{B}) \supset (\boldsymbol{A} \vee (\boldsymbol{x})\boldsymbol{B})$, if there are no free occurrences of the variable $\boldsymbol{x}$ in $\boldsymbol{A}$.

(Note that each of the meta-axioms *R5* and *R6*, like *R1–R4*, stipulates an infinity of axioms.)

The two rules of inference required are the *Rule of Modus Ponens* (*MP*) and the *Rule of Generalization* (*Gen*).

MP. If $\vdash \boldsymbol{A}$ and $\vdash \boldsymbol{A} \supset \boldsymbol{B}$, then $\vdash \boldsymbol{B}$.

Gen. If $\vdash \boldsymbol{A}$ then $\vdash (\boldsymbol{x})\boldsymbol{A}$, where $\boldsymbol{x}$ is a variable.

MP states that if $\boldsymbol{A}$ is a theorem and if '$\boldsymbol{A} \supset \boldsymbol{B}$' is a theorem, then $\boldsymbol{B}$ must be a theorem also. *Gen* states that if $\boldsymbol{A}$ is a theorem, then any result of prefixing a universal quantifier to $\boldsymbol{A}$ is also a theorem.

(It is evident that we shall have no need for a rule of substitution for individual variables. Rather than to perform such substitutions we can achieve the same effect in another way. E.g., instead of making a substitution at a particular place in a proof we can go back to those axioms or preceding theorems which contain the desired variable in the required position, and then simply retrace our steps with the desired variable already in place, rewriting, if needed, the bound variables occurring.)

It is not difficult to show that *R1–R6* together with *MP* and *Gen*, using the definitions *D1–D4* in just the form given, provide the basic theorems of the classical, two-valued theory of truth-functions and quantifiers. This will be done in § B and § C.

Many of the foregoing explanations are vague and in need of a more precise formulation. An exact analysis of the syntactical structure of logical systems of first order of the kind considered will be given in the next chapter.

B. Some Truth-Functional Theorems. The notion of being a *proof* may be defined recursively, as has already been suggested, as being a finite sequence of formulae, each formula of which is either an axiom or can be obtained from preceding formulae of the sequence by means of *MP* or *Gen*. A finite sequence of formulae,

$$A_1, A_2, \ldots, A_n,$$

constitutes a proof then provided A_i (for $1 \leq i \leq n$) is itself directly stipulated to be an axiom, or A_i may be reached from A_j and A_k ($j < i$, $k < i$, $j \neq k$) by *MP*, or A_i may be reached from A_j ($j < i$) by *Gen*. A theorem may then be thought of as the last member of a proof. In practice it is often convenient to regard as a proof also any sequence of formulae, each formula of which is either an axiom *or a previously proved theorem* or which can be reached from such by *MP* or *Gen*. It is in this somewhat broadened sense that proofs will now be given.

Strictly speaking, we shall not prove here individual theorems. We shall prove, rather, *meta-theorems*, general statements about the language stating that any formula of such and such a form is a theorem. Meta-theorems are also a kind of rule, together with meta-axioms and rules of inference. Sometimes meta-theorems or theorems will be called *laws* or *principles*.

These rough definitions of 'proof' and 'theorem' will be made precise in the next chapter. As a consequence of the definitions to be given there, the rules *MP* and *Gen* will themselves be forthcoming as theorems of syntax. (Cf. *TG7a* and *TG7b* of Chapter III.)

The theorems or meta-theorems throughout will be numbered in such a way as to indicate the section of the book in which they occur. *TB1* to follow, e.g., is the first meta-theorem in § B. Because the preceding statement itself occurs in the same chapter in which *TB1* is proved or listed, there is no need to give the chapter number also. A meta-theorem or theorem or rule proved or listed in a chapter other than that in which it is being mentioned, will have its number followed by the chapter number (in which it is proved or listed) in parentheses. E.g., *TG4*(IV) is the fourth theorem or meta-theorem of § G in Chapter IV.

Let us consider now some consequences of *R1–R4* alone, leaving aside for the moment the rules concerning quantification.[1]

The first rule or meta-theorem, a *transitivity law*, follows immediately from *R4*.

[1] Proofs are given here and throughout, not in the sense that the sequences of formulae constituting them are actually exhibited, but in the sense that explicit directions are indicated for constructing such sequences.

TB1. $\vdash (A \supset B) \supset ((C \supset A) \supset (C \supset B))$.

In *R4* we take C as '$\sim C$' and then use *D1*.

The next three rules stipulate theorems which are laws of *identity* and *excluded middle.*

TB2a. $\vdash A \supset A$.
TB2b. $\vdash \sim A \vee A$.
TB3. $\vdash A \vee \sim A$.

TB2a is proved by *R1*, *R2*, *TB1*, and *MP*. *TB2a* yields *TB2b* as an immediate consequence using *D1*. *TB3* follows from *TB2b* using *R3* and *MP*.

The kind of procedure whereby we gained *TB2a* from *R1* and *R2* by means of *TB1* is so common that it seems useful to formulate a special rule to justify this procedure. This is called a *derived rule.*

DR–TB1. If $\vdash A \supset B$ and $\vdash B \supset C$, then $\vdash A \supset C$.

This rule may be established by *TB1*. By hypothesis,

(1) $$\vdash A \supset B,$$

and

(2) $$\vdash B \supset C.$$

But by *TB1*,

(3) $$\vdash (B \supset C) \supset ((A \supset B) \supset (A \supset C)).$$

Using (2), (3), and *MP* we have that

$$\vdash ((A \supset B) \supset (A \supset C)).$$

But using this and (1) and *MP* we get that

$$\vdash A \supset C.$$

Hence we have the rule that $\vdash A \supset C$ on the hypotheses (1) and (2).

We call this rule *DR-TB1*, a derived rule based on the metatheorem *TB1*. We can also gain derived rules from *R1–R4* in similar fashion.

DR-R1. If $\vdash A \vee A$ then $\vdash A$.
DR-R2. If $\vdash A$ then $\vdash A \vee B$.
DR-R3. If $\vdash A \vee B$ then $\vdash B \vee A$.
DR-R4. If $\vdash A \supset B$ and $\vdash C \vee A$, then $\vdash C \vee B$.

The proofs are similar to that of *DR-TB1* from *TB1*, utilizing only *MP* and *R1–R4*. The use of these rules will save many steps in carrying out proofs.

The next two laws are forms of principles of *double negation.*

TB4. $\vdash A \supset \sim\sim A$.
TB5. $\vdash \sim\sim A \supset A$.

TB4 is immediate by *TB3*, putting '$\sim A$' in place of A. This gives us that

$$\vdash \sim A \vee \sim\sim A,$$

which by *D1* gives us *TB4*. To show *TB5* we first note that by *TB4*,

$$\vdash \sim A \supset \sim\sim\sim A.$$

Then using *DR-R4* and *TB3* we have that

$$\vdash A \vee \sim\sim\sim A.$$

From this, using *DR-R3* we have that

$$\vdash \sim\sim\sim A \vee A,$$

whence we get *TB5* by *D1*.

Corresponding to *TB4* and *TB5* we also have the following derived rules.

DR-TB4. If $\vdash A$ then $\vdash \sim\sim A$.
DR-TB5. If $\vdash \sim\sim A$ then $\vdash A$.

The following law is a law of *contraposition*.

TB6. $\vdash (A \supset B) \supset (\sim B \supset \sim A)$.

Using *R4* we know that

$$\vdash (B \supset \sim\sim B) \supset ((\sim A \vee B) \supset (\sim A \vee \sim\sim B)).$$

From this using *TB4* and *MP*, we know that

(1) $$\vdash (\sim A \vee B) \supset (\sim A \vee \sim\sim B).$$

But by *R3*,

(2) $$\vdash (\sim A \vee \sim\sim B) \supset (\sim\sim B \vee \sim A).$$

Hence by (1), (2), and *DR-TB1*, we get the theorem, using also *D1*.

The next two laws are *DeMorgan's laws*.

TB7. $\vdash \sim(A \,.\, B) \supset (\sim A \vee \sim B)$.
TB8. $\vdash (\sim A \vee \sim B) \supset \sim(A \,.\, B)$.

TB7 is immediate from *TB5*, because by *TB5*,

$$\vdash \sim\sim(\sim A \vee \sim B) \supset (\sim A \vee \sim B).$$

But '$\sim\sim(\sim A \vee \sim B)$' is by *D2* simply '$\sim(A \,.\, B)$'. One proves *TB8* similarly using *TB4*.

The *associative* laws for disjunction are as follows.

TB9$\vdash (A \vee (B \vee C)) \supset ((A \vee B) \vee C)$.
TB10. $\vdash ((A \vee B) \vee C) \supset (A \vee (B \vee C))$.

For the proof of *TB9* we proceed as follows.

$$\vdash C \supset (A \vee C),$$

by *R2*, *R3*, and *DR-TB1*. By this and *DR-R4* also

$$\vdash (B \vee C) \supset (B \vee (A \vee C)).$$

In a similar way we know that

$$\vdash (A \vee (B \vee C)) \supset (A \vee (B \vee (A \vee C))),$$

and, using *R3* and *DR-TB1*, that

(1) $$\vdash (A \vee (B \vee C)) \supset ((B \vee (A \vee C)) \vee A).$$

Note that by *R2*, *R3*, and *DR-TB1*,

$$\vdash (A \vee C) \supset (B \vee (A \vee C)).$$

But by *R2*,

$$\vdash A \supset (A \vee C).$$

Using these and *DR-TB1* we have that

$$\vdash A \supset (B \vee (A \vee C)).$$

Using this and *DR-R4* we know that

$$\vdash ((B \vee (A \vee C)) \vee A) \supset ((B \vee (A \vee C)) \vee (B \vee (A \vee C))).$$

But by *R1*,

$$\vdash ((B \vee (A \vee C)) \vee (B \vee (A \vee C))) \supset (B \vee (A \vee C)).$$

Using these and *DR-TB1* we know that

$$\vdash ((B \vee (A \vee C)) \vee A) \supset (B \vee (A \vee C)).$$

Combining this with (1) above we know that

(2) $$\vdash (A \vee (B \vee C)) \supset (B \vee (A \vee C)).$$

Also

(3) $$\vdash (A \vee (B \vee C)) \supset (A \vee (C \vee B)),$$

using *DR-R4* and *R3*. By (2),

$$\vdash (A \vee (C \vee B)) \supset (C \vee (A \vee B)).$$

Using now this and (3) and *DR-TB1*, we have that

(4) $$\vdash (A \vee (B \vee C)) \supset (C \vee (A \vee B)).$$

But using *R3* also

$$\vdash (C \vee (A \vee B)) \supset ((A \vee B) \vee C).$$

Hence we get the meta-theorem *TB9* by this, (4), and *DR-TB1*.

TB10, which stipulates the converses of theorems stipulated by *TB9*, is provable from *TB9* using the commutative law *R3* several times. One first notes that by *TB9*,

$$\vdash (C \vee (B \vee A)) \supset ((C \vee B) \vee A).$$

One then uses *R3*, *DR-R4*, and *DR-TB1*.

The *commutative* law for conjunction is as follows.

TB11. $\vdash (A \,.\, B) \supset (B \,.\, A)$.

By *R3* we know that

$$\vdash (\sim B \vee \sim A) \supset (\sim A \vee \sim B).$$

From this, using *DR-TB6* and *D2*, we get the meta-theorem.

Corresponding to every meta-theorem which asserts that certain formulae of conditional form are theorems,

(1) $$\vdash A \supset B,$$

we can obviously assert, in view of *MP*, a corresponding derived rule,

(2) $$\text{If } \vdash A \text{ then } \vdash B.$$

If *TBn* asserts (1) then the corresponding rule, *DR-TBn*, asserts (2). If *TBn* asserts that

(3) $$\vdash A \supset (B \supset C),$$

then *DR-TBn* can also be taken as asserting that

(4) $$\text{If } \vdash A \text{ and } \vdash B, \text{ then } \vdash C.$$

The proof is immediate from (3), using *MP* twice.

Hereafter it will not be necessary to state the derived rule or rules corresponding to a given meta-theorem. We shall be free, however, to use any such rules wherever needed. The meta-theorems to which they correspond will usually be of the forms (1) or (3) above, and the rules themselves will be of the forms (2) or (4).

Not all the derived rules needed correspond explicitly to previously proved meta-theorems. It is often convenient and useful to allow other forms of derived rule also. Especially important among these are the *Rules of Replacement*. The first such rule is as follows.

DR-Repl 1. If $\vdash A \supset B$ and $\vdash B \supset A$ then $\vdash C \supset D$ and $\vdash D \supset C$, where (*a*) C contains at least one occurrence of A, (*b*) D differs from C only in containing an occurrence of B in place of a single occurrence of A in C, and (*c*) C is of the form (i) '$\sim A$', (ii) '$(A \vee A')$', or (iii) '$(A' \vee A)$', where A does not occur in A'.

The proof is as follows. We first note that, taking C as (i), (ii), or (iii), D is then respectively

$$\text{'}\sim B\text{'}, \text{ '}(B \vee A')\text{', or '}(A' \vee B)\text{'}.$$

But on the hypotheses that $\vdash A \supset B$ and $\vdash B \supset A$ we can show that

$$\vdash \sim A \supset \sim B,$$

and

$$\vdash \sim B \supset \sim A,$$

by *DR-TB6*;

$$\vdash (A' \vee A) \supset (A' \vee B),$$

and

$$\vdash (A' \vee B) \supset (A' \vee A),$$

by *DR-R4*;

$$\vdash (A \vee A') \supset (B \vee A'),$$

and

$$\vdash (B \vee A') \supset (A \vee A'),$$

by the preceding steps, *R3*, and *DR-TB1*. Hence in each of these cases

$$\vdash C \supset D \text{ and } \vdash D \supset C$$

on the hypotheses. Hence the rule.

We now note that the hypothesis (*c*) in *DR-Repl 1* is not required. This may perhaps best be shown by means of an example. Suppose C is of the form

$$\text{`}\sim(\sim A \vee (A_2 \vee A_3))\text{'},$$

where A does not occur in A_2 or A_3. D is then

$$\text{`}\sim(\sim B \vee (A_2 \vee A_3))\text{'}.$$

Now C is not of the forms (i), (ii), or (iii) as in the hypothesis (*c*) of *DR-Repl 1*. Nonetheless we can show that

$$\vdash C \supset D \text{ and } \vdash D \supset C$$

as follows. By *DR-Repl 1*, on the hypotheses,

(1) $$\vdash \sim A \supset \sim B,$$

and

(2) $$\vdash \sim B \supset \sim A.$$

Using *DR-Repl 1* over again, on the hypotheses (1) and (2), we know that

(3) $$\vdash (\sim A \vee (A_2 \vee A_3)) \supset (\sim B \vee (A_2 \vee A_3)),$$

and

(4) $$\vdash (\sim B \vee (A_2 \vee A_3)) \supset (\sim A \vee (A_2 \vee A_3)).$$

Using *DR-Repl 1*, once again, taking (3) and (4) now as the hypotheses, we have that

$$\vdash C \supset D \text{ and } \vdash D \supset C.$$

Thus we can drop the hypothesis by iterated applications of the rule. Also the hypothesis (*a*) may obviously be dropped.

Let *DR-Repl 2* be *DR-Repl 1* with the hypotheses (*a*) and (*c*) dropped. The proof is simply the extension of the argument above to the general case. (This extension of the proof is by an *inductive* or recursive procedure in the meta-language. See (III,B).)

The third replacement rule generalizes the second to allow *zero or more* occurrences of A in C to be replaced by occurrences of B.

DR-Repl 3. If $\vdash A \supset B$ and $\vdash B \supset A$, then $\vdash C \supset D$ and $\vdash D \supset C$, where D differs from C only in containing occurrences of B in place of some or all occurrences (if any) of A in C.

The proof is not difficult now that we have *DR-Repl 2*. Suppose C contains just n ($n \geqq 0$) occurrences of A. Let D_1 differ from C if at all only in containing B in place of the left-most occurrence of A in C. Then, on the hypotheses, by *DR-Repl 2*,

$$\vdash C \supset D_1 \text{ and } \vdash D_1 \supset C.$$

Let D_2 differ from D_1 if at all only in containing B in place of the left-most occurrence of A in D_1. Then also, on the hypotheses, by *DR-Repl 2*,

$$\vdash D_1 \supset D_2 \text{ and } \vdash D_2 \supset D_1.$$

And so on until we reach some D_n which is D. Then using *DR-TB1* several times, we finally get that

$$\vdash C \supset D \text{ and } \vdash D \supset C.$$

Hence the rule.

Although the proof of *DR-Repl 3* may seem laborious, its use will facilitate many subsequent proofs. All the labor, so to speak, has been packed into the proof of this rule and of *DR-Repl 1* and *DR-Repl 2*. The remaining proofs are now easy.

Using the third replacement rule, the following are readily provable, using *TB4* and *TB5*.

TB12. $\vdash \sim(A \vee B) \supset (\sim A \,.\, \sim B)$.
TB13. $\vdash (\sim A \,.\, \sim B) \supset \sim(A \vee B)$.

Also we can prove the *associative* laws for conjunction.

TB14. $\vdash (A \,.\, (B \,.\, C)) \supset ((A \,.\, B) \,.\, C)$.
TB15. $\vdash ((A \,.\, B) \,.\, C) \supset (A \,.\, (B \,.\, C))$.

The following two laws are laws of *simplification* for conjunction.

TB16. $\vdash (A \,.\, B) \supset A$.
TB17. $\vdash (A \,.\, B) \supset B$.

The proofs are very easy. These laws enable us, so to speak, to break a conjunction into its components. The following law enables us to build up a conjunction from its components.

TB18. $\vdash A \supset (B \supset (A \,.\, B))$.

The proof is by *TB3* and *TB10*.

A derived rule corresponding to *TB18* is as follows.

DR-TB18. If $\vdash A$ and $\vdash B$ then $\vdash A \,.\, B$.

Also we have an immediate corollary of *TB18*, that

TB19. $\vdash (A \supset B) \supset ((B \supset A) \supset (A \equiv B))$.

A derived rule corresponding to *TB19* states that if

$$\vdash A \supset B \text{ and } \vdash B \supset A,$$

then

$$\vdash A \equiv B.$$

Hence from *TB4* and *TB5*, the two laws of double negation, we get immediately, using *DR-TB19*, that

TB20a. $\vdash A \equiv \sim\sim A$.

In a similar way we can prove the following equivalences.

TB20b. $\vdash (A \vee B) \equiv (B \vee A)$.
TB20c. $\vdash ((A \vee B) \vee C) \equiv (A \vee (B \vee C))$.
TB20d. $\vdash ((A \,.\, B) \,.\, C) \equiv (A \,.\, (B \,.\, C))$.
TB20e. $\vdash A \equiv (A \vee A)$.
TB20f. $\vdash A \equiv (A \,.\, A)$.

Another useful replacement rule is as follows, stated in terms of equivalence.

DR-Repl 4. If $\vdash A \equiv B$, then $\vdash C \equiv D$, where (etc., as in *DR-Repl 3*).

Also the following replacement rule is sometimes useful.

DR-Repl 5. If $\vdash A \equiv B$ and $\vdash C$, then $\vdash D$, if (etc., as in *DR-Repl 3*).

The following law is now easily provable.

TB21. $\vdash (\sim A \supset B) \equiv (A \vee B)$.

By *DR-Repl 4* and *TB20a*.

We now merely list several further meta-theorems for possible future reference.

TB22. $\vdash (A \supset (B \supset C)) \equiv ((A \,.\, B) \supset C)$.
TB23. $\vdash (A \supset B) \equiv (A \supset (A \supset B))$.
TB24. $\vdash (A \vee (B \,.\, C)) \equiv ((A \vee B) \,.\, (A \vee C))$.
TB25. $\vdash (A \,.\, (B \vee C)) \equiv ((A \,.\, B) \vee (A \,.\, C))$.
TB26. $\vdash ((A \supset B) \,.\, (A \supset C)) \equiv (A \supset (B \,.\, C))$.
TB27. $\vdash ((A \supset B) \,.\, (C \supset B)) \equiv ((A \vee C) \supset B)$.
TB28. $\vdash ((A \supset B) \vee (A \supset C)) \equiv (A \supset (B \vee C))$.
TB29. $\vdash (A \supset (B \equiv C)) \equiv ((A \supset B) \equiv (A \supset C))$.
TB30. $\vdash ((A \supset B) \,.\, (C \supset D)) \supset ((A \,.\, C) \supset (B \,.\, D))$.
TB31. $\vdash ((A \supset B) \,.\, (C \supset D)) \supset ((A \vee C) \supset (B \vee D))$.
TB32. $\vdash (A \,.\, (A \supset B)) \supset B$.
TB33. $\vdash (A \,.\, \sim A) \supset B$.

C. Theorems of Quantification. Let us now investigate some of the consequences of the Rules of Quantification, the most important of which are as follows.

TC1. $\vdash B \supset (\mathrm{E}x)A$, if B differs from A only in containing free occurrences of some one variable or term wherever there are free occurrences of the variable x in A.

The proof is as follows. By *R5*,

$$\vdash (x)\sim A \supset \sim B,$$

if (etc.). (Note that if B differs from A in the way described in *R5*, then '$\sim B$' differs from '$\sim A$' in the same way.) Hence by *DR-TB6*,

$$\vdash \sim\sim B \supset \sim(x)\sim A,$$

if (etc.). Hence *TC1*, using *D4*, *TB4*, and *DR-TB1*.

Henceforth we shall frequently omit reference in the proofs to *R1–R4*, to *MP*, or to any of the meta-theorems or rules of § B. The way in which these meta-theorems or rules are being used will be obvious enough. Also, unless otherwise stipulated, 'x', 'y', and 'z' are to be regarded in this section as ranging over just variables, rather than terms.

Using *TC1* and *R5*, we have that

TC2. $\vdash (x)A \supset (\mathrm{E}x)A$.

The converses of theorems stipulated by *R6* are provable as follows.

TC3. $\vdash A \vee (x)B \,.\!\supset\!.\, (x)(A \vee B)$, if x does not occur freely in A.

Clearly

$$\vdash A \vee (x)B \,.\!\supset\!.\, A \vee B,$$

by *R5* and *DR-R4*. Hence by *Gen*,

(1) $$\vdash (x)(A \vee (x)B \,.\!\supset\!.\, A \vee B).$$

But by *R6*, also

(2) $$\vdash (x)(A \vee (x)B \,.\!\supset\!.\, A \vee B) :\!\supset\!: A \vee (x)B \,.\!\supset\!.\, (x)(A \vee B),$$

if x does not occur freely in '$A \vee (x)B$'. But if x does not occur freely in A then it cannot occur freely in '$A \vee (x)B$' either. Hence, using (1), (2), and *MP*, we get *TC3*.

Combining *TC3* with *R6* we have that

TC4. $\vdash (x)(A \vee B) \,.\!\equiv\!.\, A \vee (x)B$, if there are no free occurrences of x in A.

We can form derived rules from *R5*, *R6*, *TC1*, *TC2*, or *TC3* just as we did in § B, the theorems stipulated by these rules being of conditional form. *DR-TC2*, e.g., reads:

$$\text{If } \vdash (x)A, \text{ then } \vdash (\mathrm{E}x)A.$$

Subsequent proofs will be facilitated somewhat if we allow ourselves to use such rules wherever needed.

The following meta-theorem is immediate, from *TC4*, using *D1*.

TC5. $\vdash (x)(A \supset B) .\equiv. A \supset (x)B$, if x is not free in A.

Corresponding to *TC5*, stipulating equivalences, we have *two* derived rules, one that if

$$\vdash (x)(A \supset B),$$

where x is not free in A, then

$$\vdash A \supset (x)B,$$

on the same hypothesis. The other rule states the converse of this. Either of these rules may be referred to as *DR-TC5*.

The following law enables us to *distribute* a quantifier over a conjunction.

TC6. $\vdash (x)(A . B) :\equiv: (x)A . (x)B$.

The theorems stipulated by this rule being equivalences or mutual conditionals, we divide the proof into two parts and first prove the conditionals from left to right. By *R5*,

$$\vdash (x)(A . B) :\supset: A . B.$$

Hence, using *TB16* and *DR-TB1*,

(1) $$\vdash (x)(A . B) \supset A.$$

Similarly,

(2) $$\vdash (x)(A . B) \supset B.$$

From (1), (2), *Gen*, and *DR-TC5*, we get that

$$\vdash (x)(A . B) \supset (x)A,$$

and

$$\vdash (x)(A . B) \supset (x)B.$$

Combining these we get the conditionals from left to right, i.e., that

(3) $$\vdash (x)(A . B) :\supset: (x)A . (x)B.$$

Now, to prove the converses, we have obviously that

$$\vdash (x)A \supset A,$$

and

$$\vdash (x)B \supset B.$$

Hence,

$$\vdash (x)A . (x)B :\supset: A . B.$$

By *Gen* and *DR-TC5*, we have that

(4) $$\vdash (x)A . (x)B :\supset: (x)(A . B).$$

Combining (3) and (4) we get *TC6*.

The next two rules provide principles concerning *vacuous* quantification. They tell us that a formula consisting of a quantifier '(x)'

or '(Ex)' prefixed to a formula A not containing a free occurrence of x is equivalent to A itself.

TC7a. $\vdash (x)A \equiv A$, if x is not free in A.

TC7b. $\vdash (\mathrm{E}x)A \equiv A$, if (etc., as in *TC7a*).

The following law is also a law of *distribution*. It is provable by *TC6* and *TC7a*.

TC8. $\vdash (x)(A \,.\, B) :\equiv: A \,.\, (x)B$, if there are no free occurrences of x in A.

We also have a law of distribution of the universal quantifier with respect to '$\supset$'.

TC9. $\vdash (x)(A \supset B) \,.\supset.\, (x)A \supset (x)B$.

Clearly

$$\vdash (x)(A \supset B) \,.\, (x)A :\supset: (A \supset B) \,.\, A.$$

Hence

$$\vdash (x)(A \supset B) \,.\, (x)A :\supset: B.$$

From this we get *TC9* by *Gen* and *DR-TC5*. (The converses do not hold.)

Also we have the corresponding distribution law concerning material equivalence.

TC10. $\vdash (x)(A \equiv B) \,.\supset.\, (x)A \equiv (x)B$.

The proof is by *TC9* and *TC6*.

The following laws are concerned with the interrelationship between universal and existential quantification.

TC11a. $\vdash (\mathrm{E}x)A \equiv \sim (x)\sim A$.

TC11b. $\vdash \sim(\mathrm{E}x)A \equiv (x)\sim A$.

TC11c. $\vdash (\mathrm{E}x)\sim A \equiv \sim (x)A$.

TC11d. $\vdash \sim(\mathrm{E}x)\sim A \equiv (x)A$.

The proofs are not difficult.

Related to *TC9* and *TC10* are the following meta-theorems concerning existential quantification.

TC12. $\vdash (x)(A \supset B) \,.\supset.\, (\mathrm{E}x)A \supset (\mathrm{E}x)B$.

TC13. $\vdash (x)(A \equiv B) \,.\supset.\, (\mathrm{E}x)A \equiv (\mathrm{E}x)B$.

The following rules will be listed without proof.

TC14. $\vdash (x)(A \supset B) \,.\equiv.\, (\mathrm{E}x)A \supset B$, if there are no free occurrences of x in B.

TC15. $\vdash (\mathrm{E}x)(A \supset B) \,.\equiv.\, (x)A \supset B$, if (etc., as in *TC14*).

TC16. $\vdash (x)(A \supset B) \,.\, (x)(B \supset C) :\supset: (x)(A \supset C)$.

TC17. $\vdash (x)(A \equiv B) \,.\supset.\, (x)((A \,.\, C) \equiv (B \,.\, C))$.

TC18. $\vdash (\mathrm{E}x)(A \,.\, B) :\equiv: A \,.\, (\mathrm{E}x)B$, if there are no free occurrences of x in A.

TC19. $\vdash (\mathrm{E}x)(A \vee B) :\equiv: A \vee (\mathrm{E}x)B$, if (etc., as in *TC18*).
TC20. $\vdash (\mathrm{E}x)(A \supset B) .\equiv. A \supset (\mathrm{E}x)B$, if (etc., as in *TC18*).
TC21. $\vdash (x)A \vee (x)B .\supset. (x)(A \vee B)$.

Clearly the converses of theorems stipulated by *TC21* do not hold.

TC22. $\vdash (\mathrm{E}x)(A \vee B) .\equiv. (\mathrm{E}x)A \vee (\mathrm{E}x)B$.
TC23. $\vdash (\mathrm{E}x)(A . B) :\supset: (\mathrm{E}x)A . (\mathrm{E}x)B$.

The converses of theorems stipulated by *TC23* do not hold.

TC24. $\vdash \sim (\mathrm{E}x)(A . B) \equiv (x)(A \supset \sim B)$.
TC25. $\vdash (\mathrm{E}x)A .\supset. (x)(A \supset B) \equiv B$, if there are no free occurrences of x in B.
TC26. $\vdash \sim(\mathrm{E}x)A \supset (x)(A \supset B)$.
TC27. $\vdash (\mathrm{E}x)(A \supset B) .\equiv. (x)A \supset (\mathrm{E}x)B$.
TC28. $\vdash (x)(A \vee B) .\supset. (\mathrm{E}x)A \vee (\mathrm{E}x)B$.
TC29. $\vdash (x)(A \vee B) .\supset. (x)A \vee (\mathrm{E}x)B$.
TC30. $\vdash (x)A \vee (\mathrm{E}x)B .\supset. (\mathrm{E}x)(A \vee B)$.
TC31. $\vdash (x)(y)A .\equiv. (y)(x)A$.
TC32. $\vdash (\mathrm{E}x)(\mathrm{E}y)A .\equiv. (\mathrm{E}y)(\mathrm{E}x)A$.
TC33. $\vdash (\mathrm{E}x)(y)A .\supset. (y)(\mathrm{E}x)A$.

This completes our brief survey of the basic elementary metatheorems concerning the truth-functional connectives and quantifiers. These laws will constantly be used throughout the sequel, usually without explicit mention.

For truth-functional and quantificational theorems of a more advanced nature, the reader is referred to the treatises mentioned of Church, Hilbert-Bernays, Hilbert-Ackermann, and others.

D. Abstraction. We now introduce a notation for an operation of *abstraction*.[1]

If desired we could introduce the following definition.

(Def) '$x \varepsilon y \ni B$' is an alternative notation for A, where x is a term, y is a variable not occurring freely in A, and B differs from A only in containing free occurrences of y in 0 or more places where there are free occurrences of x in A.

Although this definition would not be *sensu stricto* an abbreviation, it does introduce a convenient and useful alternative notation. E.g., it would enable us to write

$$\text{'}x \varepsilon x' \ni \mathrm{P}x\text{''} \text{ or } \text{'}x \varepsilon x' \ni \mathrm{P}'x'x\text{''}$$

[1] The word 'abstraction' is used here in a precise technical sense, and this use is not of course to be confused with that of § F of Chapter I.

as alternatives respectively to

$$\text{'P}x\text{' or 'P}'xx''\text{'},$$

or

(1) $$\text{'}x \varepsilon x'\ni(\text{P}x' \text{ .v. } \sim(\text{E}x'')\text{P}'x''x')\text{'}$$

as an alternative to

(2) $$\text{'}(\text{P}x \text{ .v. } \sim(\text{E}x'')\text{P}'x''x)\text{'}.$$

In, e.g., (1) the expression

$$\text{'}x'\ni(\text{P}x' \text{ .v. } \sim(\text{E}x'')\text{P}'x''x')\text{'}$$

is *explicitly exhibited as a predicate*, thereby being in effect an expression for a property ascribed to x in (2). A better way of writing (1) perhaps, more in accord with the convention of writing 'Px' rather than '$x \varepsilon$ P', would be

$$\text{'}x'\ni(\text{P}x' \text{ .v. } \sim(\text{E}x'')\text{P}'x''x')\ (x)\text{'}.$$

And, because no ambiguity will result, one can drop the parentheses around the right-most occurrence of 'x'. Or, more generally, an alternative way of writing the definiendum of (Def) above would be

$$\text{'}\boldsymbol{y}\ni\boldsymbol{B}\ \boldsymbol{x}\text{'}.$$

The '$\ni$' is Peano's symbol for class abstraction.[1] The usage here is so similar to that of Peano as to justify the notation. Let any expression of the form '$\boldsymbol{y}\ni\boldsymbol{B}$' be called a *one-place abstract*. The use of abstracts will be especially helpful and convenient in discussing rigorously some of the semantical properties of first-order languages below (Chapter IV *et seq.*).

Note that in virtue of (Def) a notation for abstracts need in no way be taken as part of the primitive notation. Such expressions can always be eliminated at will. Although theoretically we can quite well do without them, in practice much is gained if we use them rather extensively. In the subsequent development, in fact, fundamental use will be made of them. *Abstracts containing no free variables* (together with primitive, one-place predicate constants) *will be taken as the fundamental kinds of expressions which denote.* Therefore, rather than to introduce the definition (Def) above, we prefer to include a notation for abstraction among the primitives of the language-system.

To the logical primitives let us then add '$\ni$', and in the definition of 'formula' let us incorporate the following clause.

5. If $\boldsymbol{A}$ is a formula, $\boldsymbol{x}$ is a term, and $\boldsymbol{y}$ is a variable, then '$\boldsymbol{y}\ni\boldsymbol{A}\ \boldsymbol{x}$' is a formula.

[1] Cf. G. Peano, *Formulaire de Mathématiques* (Gauthier-Villars, Paris: 1901), p. 1.

And to the rules *R1–R6* let us add the following rule, stipulating *Axioms of Abstraction.*

Abst. $\vdash y \ni B\,x \,.\equiv.\, A$, if (etc., as in the definition (Def) above).

The logical rules hereafter are to be thought of as comprising *R1–R6*, *MP*, *Gen*, and *Abst.*

An abstract '$x \ni A$' where there are no free occurrences of x in A is called a *vacuous* abstract; otherwise, *non-vacuous.*

Occurrences of x in A in the context '$x \ni A$' must now be regarded as *bound* occurrences, so that the definitions of *bound occurrence* and *free occurrence* given above in § A must be emended slightly.

One-place abstracts which contain no free variables behave very much like primitive, one-place predicate constants. We shall frequently refer to them therefore as *defined* predicate constants. The predicate constants comprise then both the primitive and the defined predicate constants.

The question might be asked as to the kind of entity defined predicate constants stand for. These entities we shall call *pseudo-classes* or (following Quine) *virtual classes* of objects of the fundamental domain. E.g.,

$$'x \ni (\mathrm{P}x \,.\mathrm{v}.\, \sim(\mathrm{E}x'')\mathrm{P}'x''x)'$$

may be thought of as standing for the totality of those objects which have the property P or to which no object bears the relation P′. This totality of objects is not strictly a class. Classes (relative to a given language) are the kind of entity which can significantly be taken as values for the class variables of that language, assuming it to contain such over and above individual variables. But the languages being discussed contain no such variables. Such totalities are therefore merely virtual. (See (IV,C).)

We let hereafter 'F', 'G', and 'H' be variables (of the informal meta-language) ranging over the one-place primitive predicate constant(s) and over the one-place abstracts containing no free variables. Also 'x', etc., are to be taken hereafter as terms, unless the context requires that they be taken only as variables.

What may be called *identity* of virtual classes is definable as follows.

$$'F = G' \text{ abbreviates } '(x)(Fx .\equiv. Gx)'.$$

Two virtual classes are identical if and only if the members of one are just the members of the other.

Several other notions concerned with virtual classes could also be defined here, such as the *inclusion* of two virtual classes, the *logical sum* of two virtual classes, etc. But such notions are not needed for present purposes.

In addition to *Abst*, we have the following laws concerning abstracts.

TD1. $\vdash x \ni A\, y \,.\equiv.\, A$, if '$x \ni A$' is vacuous.

TD2a. $\vdash x \ni (A \vee B)\, y :\equiv: x \ni A\, y \,.\vee.\, x \ni B\, y$.

TD2b. $\vdash x \ni \sim A\, y \,.\equiv.\, \sim x \ni A\, y$.

TD2c. $\vdash x \ni (y) A\, z \,.\equiv.\, (y) x \ni A\, z$, if x and y are distinct variables and z is any term in which y does not occur freely.

TD3a. If $\vdash A$ then $\vdash y \ni A\, x$.

TD3b. If $\vdash \sim A$ then $\vdash \sim y \ni A\, x$.

TD3a holds by *TD1* if '$y \ni A$' is vacuous. Suppose therefore that '$y \ni A$' is not vacuous. Let A differ from some B only in containing free occurrences of a variable y in one or more places where there are free occurrences of some term x in B, y not occurring freely in B. Then if $\vdash A$, $\vdash (y) A$ by *Gen*, and hence by *R5* and *MP*, $\vdash B$. But then by *Abst* $\vdash y \ni A\, x$ also. The proof of *TD3b* is similar.

We have been speaking only of *one-place* abstracts. One might wish also to introduce (relational) abstracts of more than one place (of higher degree). For many of the semantical investigations of this book, the interesting fact is that there will be no need for abstracts of higher degree at all. But if they are needed they may easily be introduced. For any $n \geqq 2$, we can define expressions such as

$$`x_1 \ldots x_n \,\varepsilon\, y_1 \ldots y_n \ni X'$$

or

$$`y_1 \ldots y_n \ni X\, x_1 \ldots x_n'$$

essentially as above. The principle *Abst* may then be extended so as to include these relational abstracts as well. For most subsequent purposes we shall not need relational abstracts, and hence we shall need only the restricted *Abst* as given above.

E. Identity. The relation of identity plays so important a role in formalized language-systems that it is frequently convenient to take a symbol for it as a primitive predicate constant. In fact the usual symbol for identity, '=', could be used in place of, say, the two-place primitive 'P''. And then we could write '$=xx$'' in the more familiar way as '$x = x$''. This would be one way of handling identity within the framework of the language-systems being discussed.

Another way of introducing identity is to take '=' as an additional logical constant. In the definition of 'formula' one then incorporates a clause to the effect that

$$`x = y'$$

is a formula wherever x and y are terms. And one adds two rules

stipulating the *Axioms of Identity*, i.e.,

R7. $\vdash x = x$,
R8. $\vdash x = y \supset (Fx \supset Fy)$.[1]

The second method has the advantage perhaps of greater explicitness because of the presence of *R7* and *R8* as rules of the underlying logic. The extended systems, including *R7* and *R8*, are sometimes called languages of first order *with identity*. We shall employ such extended systems in later chapters, so that identity will be handled throughout (unless otherwise indicated) by the second method. (See, however, (VI,A) and (XI,A).)

From *R7* and *R8*, presupposing *R1–R6*, *MP*, *Abst*, and *Gen*, one can prove the basic meta-theorems providing for the theory of identity. Some of these may be listed as follows.[2]

TE1. $\vdash x = y \supset (x = z \supset y = z)$.
TE2. $\vdash x = y \supset y = x$.
TE3. $\vdash \sim x = y \supset \sim y = x$.
TE4. $\vdash x = y \supset (y = z \supset x = z)$.
TE5. $\vdash x = z \supset (y = z \supset x = y)$.
TE6. $\vdash \sim x = y \supset (\sim x = z \vee \sim y = z)$.

In *TE1–TE6* x and y may be taken as any terms.

In the following two identity rules, *TE7* and *TE8*, y may be any term and x may be any variable not occurring in y. In *TE9* and *TE10*, y may be taken as any term and then x and z may be any distinct variables and x may not occur freely in y. Finally, in *TE11* and *TE12*, x and y may be taken as any distinct variables.

TE7. $\vdash Fy \equiv (x)(x = y \supset Fx)$.
TE8. $\vdash Fy \equiv (\mathrm{E}x)(x = y \,.\, Fx)$.
TE9. $\vdash (x)(\sim x = y \supset Fx) \equiv ((x)Fx \vee (\sim Fy \,.\, (x)(z)(\sim x = z \supset (Fx \vee Fz))))$.
TE10. $\vdash (\mathrm{E}x)(\sim x = y \,.\, Fx) \equiv ((\mathrm{E}x)Fx \,.\, (Fy \supset (\mathrm{E}x)(\mathrm{E}z)(\sim x = z.\, Fx \,.\, Fz)))$.
TE11. $\vdash (\mathrm{E}x)(y)(\sim y = x \supset Fy) \equiv (x)(y)(\sim x = y \supset (Fx \vee Fy))$.
TE12. $\vdash (x)(\mathrm{E}y)(\sim y = x \,.\, Fy) \equiv (\mathrm{E}x)(\mathrm{E}y)(\sim x = y \,.\, Fx \,.\, Fy)$.

The proofs are left for the reader.

We have now the framework for an infinity of different language-systems, depending upon the individual, predicate, or functional constants taken as primitive, and upon whether '=' is taken as an additional logical primitive. Let '*L*' be a variable hereafter ranging

[1] *R8* can be considerably weakened, if desired, by requiring that Fx be equivalent (by *Abst*) to an atomic formula.

[2] Cf. Hilbert and Bernays, *op. cit.*, vol. I, pp. 164–77.

over this infinity of systems. All the languages L are known as *simple, applied* language-systems of first order with or without identity.

There are several alternative ways of formulating these languages of first order. They may, e.g., be formulated so as to contain an entirely new kind of variable, taking properties or classes as values, although quantification over such would not be admitted. A formulation including class variables is somewhat more complicated than that used here. A *simple* language of first order is one which contains no such variables, a *non-simple* one containing such. A simple, *applied* language-system of first order is one which is simple and which (i) contains at least one predicate constant, or (ii) contains identity and at least one functional constant.[1] Ordinarily such languages do contain at least one predicate constant. But where identity is present, primitive functors often enable one to achieve equal expressive power. All the language-systems we shall be concerned with in this book in any detail are simple, applied languages of first order with or without identity in the sense of (i) or with identity in the sense of (ii). The presence of primitive individual constants is optional.

Only the logical framework of L has been sketched above. But in addition to the axioms stipulated by the logical rules *R1–R8*, L will also contain some non-logical axioms and perhaps also some non-logical rules of inference. These axioms and rules will characterize the properties, functions, and individuals designated by the primitive constants. Several different sets of such axioms and rules will be given in § L below and in later chapters.

F. Singular and Selective Descriptions. Two important notions may now be introduced within any L, that of a *singular* and of a *selective* description.

By a selective description is meant roughly a phrase of the form 'any selected or representative object x having such and such a property'. Such phrases are both *selective* and *descriptive* in the sense that some one object is picked out and described. This notion is an important one, but one which has not been used very much by philosophers or methodologists.

Selective descriptions are introduced by means of Hilbert's ε-function,[2] which may be defined in context as follows.

[1] Cf. A. Church, *op. cit.*, pp. 33–8. The terminology used here differs slightly from that of Church as well as from that of Hilbert and Ackermann.

[2] Hilbert and Bernays, *op. cit.*, vol. 2 (Berlin, Springer: 1939), pp. 9–18. This usage of 'ϵ' is of course not to be confused with that of § D above nor with those introduced in Chapter VI below.

'$(\varepsilon \boldsymbol{x}.\boldsymbol{Fx})[\boldsymbol{A}']$' abbreviates '$((\mathrm{E}\boldsymbol{x})\boldsymbol{Fx} \,.\, (\boldsymbol{x})(\boldsymbol{Fx} \,.\supset.\, \boldsymbol{A}))$', if $\boldsymbol{A}'$ differs from $\boldsymbol{A}$ only in containing '$(\varepsilon \boldsymbol{x}.\boldsymbol{Fx})$' wherever there are free occurrences of $\boldsymbol{x}$ in $\boldsymbol{A}$.[1]

According to this definition,

(1) '$(\varepsilon x.\mathrm{P}x)[\mathrm{P}x' \,.\mathrm{v}.\, \mathrm{P}'(\varepsilon x.\mathrm{P}x)x']$',

e.g., is short for

(2) '$(\mathrm{E}x)\mathrm{P}x \,.\, (x)(\mathrm{P}x \,.\supset.\, (\mathrm{P}x' \,.\mathrm{v}.\, \mathrm{P}'xx'))$'.

An expression such as '$(\varepsilon x.\mathrm{P}x)$' may be read 'any selected x having P'. That this reading is perhaps justified may be seen as follows. (2) above we should read naturally as 'There is at least one object which has P and everything x which has P is such that x' has P or x bears P′ to x''. If we use 'any' in such a way that the phrase 'any selected so-and-so' involves the existence of at least one so-and-so, then (1) may be read 'Any selected object x having P is such that Px' or x bears P′ to x'', where it is understood that there is at least one object having P. The use of 'any' in this way seems, to some extent at least, in accord with its customary usage.

In an expression of the form '$(\varepsilon \boldsymbol{x}.\boldsymbol{Fx})[\boldsymbol{A}]$', the first or left-most occurrence of '$(\varepsilon \boldsymbol{x}.\boldsymbol{Fx})$' is called a *scope indicator*. We should note especially why the scope indicators are important. They are needed here just as in Russell's theory of singular descriptions below and for similar reasons. Where $\boldsymbol{A}$ is a sentence of L, $\boldsymbol{F}$ contains no free variables, and '$\sim(\mathrm{E}x)\boldsymbol{F}x$' is true, we must be careful to distinguish between

(3) '$(\varepsilon x.\boldsymbol{F}x)[\boldsymbol{G}(\varepsilon x.\boldsymbol{F}x)] \,.\supset.\, \boldsymbol{A}$'

and

(4) '$(\varepsilon x.\boldsymbol{F}x)[\boldsymbol{G}(\varepsilon x.\boldsymbol{F}x) \,.\supset.\, \boldsymbol{A}]$'.

Because, where '$\sim(\mathrm{E}x)\boldsymbol{F}x$' is true, (3) is also true whereas (4) is false. This is seen if we expand (3) and (4) according to the definition. (3) becomes

$$'(\mathrm{E}x)\boldsymbol{F}x \,.\, (x)(\boldsymbol{F}x \,.\supset.\, \boldsymbol{G}x) :\supset: \boldsymbol{A}',$$

which is true, the hypothesis being false. On the other hand, (4) becomes

$$'(\mathrm{E}x)\boldsymbol{F}x \,.\, (x)(\boldsymbol{F}x :\supset: \boldsymbol{G}x \,.\supset.\, \boldsymbol{A})',$$

which is false if '$\sim(\mathrm{E}x)\boldsymbol{F}x$' is true. Thus it is essential to distinguish between (3) and (4) by placing the scope indicators appropriately.

[1] This definition in essential respects appears due to Professor Frederic B. Fitch.

Wherever extensive use is made of the ε-function, the following definition is useful.

'$\mathrm{E}!(\varepsilon x.Fx)$' is an alternative notation for '$(\mathrm{E}x)Fx$'.

To say that $\mathrm{E}!(\varepsilon x.\mathrm{P}x)$, e.g., is to say that there is at least one object having P.

The two basic meta-theorems governing the ε-function are the following.

TF1. $\vdash (\mathrm{E}x)Fx \,.\equiv.\, (\varepsilon x.Fx)[F(\varepsilon x.Fx)]$,

and

TF2. $\vdash (x)Fx \,.\equiv.\, \sim(\varepsilon x.\sim Fx)[\sim F(\varepsilon x.\sim Fx)]$.

The proofs are immediate. These meta-theorems show how existential and universal quantification, respectively, can be expressed exclusively in terms of the Hilbert 'ε'.

Closely related to the notion of a selective description is that of a singular description.[1] A singular description is, speaking roughly, a phrase of the form 'the so-and-so' or 'the one so-and-so'. Within *L*, if identity is present, a notation for singular descriptions may be introduced as follows.

'$(\imath x.Fx)[A']$' abbreviates '$((\varepsilon x.Fx)[A] \,.\, (x)(y)(Fx \,.\, Fy :\supset: x = y))$', if x and y are distinct variables and A' differs from A only in containing occurrences of '$(\imath x.Fx)$' wherever and only where A contains occurrences of '$(\varepsilon x.Fx)$'.

The expression '$(\imath x.\mathrm{P}x)$', e.g., may be read 'the one x having P'. The scope indicators are required here just as above in the case of selective descriptions.

Wherever extensive use is made of singular descriptions the following definition is also useful.

'$\mathrm{E}!(\imath x.Fx)$' for '$(\mathrm{E}y)(x)(x = y \,.\equiv.\, Fx)$', where x and y are distinct variables.

The basic theorems concerning singular descriptions may be listed as follows.

TF3. $\vdash \mathrm{E}!(\imath x.Fx) :\equiv: (\mathrm{E}x)Fx \,.\, (x)(y)(Fx \,.\, Fy :\supset: x = y)$, if x and y are distinct variables.

TF4. $\vdash (\imath x.Fx)[A'] :\equiv: \mathrm{E}!(\imath x.Fx) \,.\, (x)(Fx \,.\supset.\, A)$, if A' differs from A only in containing '$(\imath x.Fx)$' wherever there are free occurrences of x in A.

[1] Cf. Whitehead and Russell, *op. cit.*, vol. 1, pp. 172–86. Also Hilbert and Bernays, *op. cit.*, vol. 1, pp. 383 *et seq.* For the philosophical importance of description theory, see *inter alia* B. Russell, *Introduction to Mathematical Philosophy* (George Allen and Unwin, London: 1919), pp. 167–80.

TF5. $\vdash (\iota x.Fx)[(\iota y.Gy)[A]] .\equiv. (\iota y.Gy)[(\iota x.Fx)[A]]$, if x and y are distinct variables.

Singular descriptions are useful primarily as a means for defining individual constants. Suppose L contains no primitive individual constants. And suppose the axioms of L are such as to assure the existence of one and only one object x such that ---x---. We can then define

'ā' as '$(\iota x.$---x---$)$'.

Hence if L contains primitively no individual constants they may be introduced by means of descriptions if the requisite primitive predicates are available. And, conversely, if L contains individual constants as primitive, these may be dropped and defined by means of descriptions if the requisite primitive predicates are available. These important facts enable us to see that an L containing primitive individual constants can easily be transformed into an L' containing none, and conversely, under suitable circumstances.

Using identity, singular descriptions, and abstracts of higher degree, we can also eliminate *functional* constants as primitives of L under suitable circumstances. Suppose Q is a primitive or defined $(n+1)$-place predicate constant of L and that

$$\vdash (x_1) \ldots (x_{n+1})(x_{n+2})(Qx_1x_2 \ldots x_{n+1} \;.\; Qx_{n+2}x_2 \ldots x_{n+1} :\supset: x_1 = x_{n+2}),$$

and $$\vdash (x_1) \ldots (x_n)(\mathrm{E}x_{n+1})Qx_{n+1}x_1 \ldots x_n.$$

Then we may define

$$\text{'}(\lambda x_1 \ldots x_n.Q)\text{' as '}yx_1 \ldots x_n \ni Qyx_1 \ldots x_n\text{'}.$$

But in turn an n-place functional constant, say 'f̄', may be regarded as an abbreviation for

$$\text{'}(\lambda x_1 \ldots x_n.Q)\text{'},$$

and the term

$$\text{'}\bar{f}y_1 \ldots y_n\text{' may then abbreviate '}(\iota x_{n+1}.\ \bar{f}x_{n+1}y_1 \ldots y_n)\text{'}.$$

Hence if desired we may eliminate functional constants as primitives of L under suitable circumstances, and introduce them by means of these definitions.

Because individual and functional constants may always be eliminated as primitives (under suitable circumstances) with no loss of expressive power, as we have just seen, we shall be justified in many of our subsequent investigations if we confine attention to those L's containing none such primitively.

G. Normal Forms. Two theorems concerning the language-systems L will be especially useful for the semantical investigations

below. These tell us that the formulae of L can be brought into certain standard *normal forms*. There are of course other normal forms also but the two that will be especially useful for what follows are a kind of *conjunctive* normal form and the *prenex* normal form.[1]

Let any formula of L of the kind '$(x)A$', where A contains no free variable other than x and may or may not contain x itself as a free variable, be called an *atomic universal sentence*. Such a sentence is universal because of the presence of the universal quantifier. But it is also atomic in the sense that it can enter, by means of negation and disjunction, into the composition of longer molecular formulae. E.g., '$(x)A$' can be regarded as an atomic universal part of '$\sim(x)A \supset B$'.

We may suppose for the moment that L contains no primitive individual constants, so that any *sentence* of L will contain a quantifier and will be either an atomic universal sentence or constructed out of such by means of '$\sim$' and 'v' in the appropriate way. Given any sentence C of L we can then construct a certain sentence out of its constituent atomic universal sentences as follows. Let $A_1, \ldots, A_n$ be the constituent atomic universal sentences of C, and let B_i be either A_i or '$\sim A_i$' ($1 \leq i \leq n$). We then form various disjunctions of the B's and finally a conjunction of these disjunctions. Any such conjunction of disjunctions of atomic universal sentences or their negations will be said to be in *special conjunctive normal form*.

A theorem concerning this conjunctive normal form is as follows.

Given any formula C of L, containing no free variables, there exists a formula D in special conjunctive normal form such that $\vdash C \equiv D$.

The proof uses merely theorems of truth-functional logic.

Let us consider an example or two. Let A_1, A_2, and A_3 be atomic universal sentences. Consider the formula

(1) '$(((A_1 \supset (A_2 \supset A_3)) \,.\, A_1 \,.\, A_2) \supset A_3$'.

By suitable transformations, using some consequences of *R1–R4*, one can show this equivalent to

(2) '$(A_1 \vee \sim A_1 \vee \sim A_2 \vee A_3) \,.\, (A_2 \vee \sim A_1 \vee \sim A_2 \vee A_3) \,.\, (\sim A_3 \vee \sim A_1 \vee \sim A_2 \vee A_3)$'.

Note that (2) is in special conjunctive normal form. *Pari passu*,

'$(A_1 \vee A_2 \vee \sim A_1) \,.\, (\sim A_2 \vee A_2 \vee \sim A_1) \,.\, (\sim A_2 \vee \sim A_1 \vee A_1) \,.\, (A_1 \vee \sim A_1 \vee A_2)$'

[1] Cf., e.g., Hilbert and Ackermann, *op. cit.*, pp. 10–11 and p. 67, or A. Church, *op. cit.*, pp. 59–61.

is in special conjunctive normal form and is equivalent to

$$'(A_1 \supset A_2) \supset (\sim A_2 \supset \sim A_1)'.$$

Another important normal form is the so-called prenex normal form, concerned with the quantifiers. A formula C of L is in prenex normal form if it is of the form

$$(3) \qquad '(Qx_1)(Qx_2) \ldots (Qx_n)A',$$

where each '(Qx_i)' is either '(x_i)' or '(Ex_i)' for $1 \leq i \leq n$, A contains no quantifiers, and $x_1, \ldots, x_n$ are distinct variables occurring freely in A.

The *Prenex Normal Form Theorem* is as follows.

Given any formula C of L, containing no free variables, there exists a formula D of L in prenex normal form such that $\vdash C \equiv D$.

The proof requires primarily some elementary laws of quantification. To take an example, it is easy to show that the formula

$$'(x)(A \supset B) \supset ((Ey)C_1 \supset (Ez)C_2)'$$

has

$$'(Ex)(y)(Ez)((A \supset B) \supset (C_1 \supset C_2))'$$

as a prenex normal form, provided x is not free in C_1 or C_2, y is not free in A, B, or C_2, and z is not free in C_1, A, or B. And under similar provisos,

$$'(Ex)(A \,.\, B) \supset (y)(C_1 \supset (Ez)C_2)'$$

has as a prenex normal form

$$'(x)(y)(Ez)((A \,.\, B) \supset (C_1 \supset C_2))'.$$

In both these examples, x, y, and z are to be taken as distinct variables.

Note that the definitions here do not require that the normal forms concerned be unique. On the contrary, given any sentence of L, there may be several formulae in conjunctive or prenex normal form equivalent to it.

Both of these normal forms will be useful in the sequel in connection with the definition of the semantical truth-concept for L. In the case of the prenex normal form, we find it useful to require also that the left-most quantifier of a formula in prenex normal form be a universal quantifier. This can always be achieved easily. Suppose the left-most quantifier '(Qx_1)' is '(Ex_1)' and that '$=$' is available. We then form from the formula (3) in prenex normal form the formula

$$'(x_{n+1})(Ex_1)(Qx_2) \ldots (Qx_n)(A \,.\, x_{n+1} = x_{n+1})'.$$

This formula is obviously equivalent with (3), taking x_{n+1} as a new variable. In this way we can assume without loss of generality that

all formulae in prenex normal form contain universal quantifiers as their left-most quantifiers.

H. Domains of Individuals. Let us now turn to some considerations of a somewhat less technical character.

One of the dominant characteristics which determines a language-system L to be of first order is that its variables are regarded as ranging over a single kind of object, called *individuals*, as we have seen. Of course there is great latitude as to what may be regarded as individuals. The totality of space-time points, the totality of natural numbers, the totality of protons and electrons, the totality of human beings, etc., may be taken as constituting a domain of individuals for appropriate L. But even objects of a more "abstract" kind may be regarded as individuals. The totality of real numbers, that of real numbers together with all classes or functions of real numbers, the totality of objects of set theory, the totality of art objects (in some well-defined sense), etc., may all appropriately be regarded as constituting a domain of individuals for a suitably formulated L. In fact, any kind of entity whatsoever may be taken as a domain of individuals, provided only that it be a clear-cut, well-demarcated totality.

What is here called a 'domain of individuals' is sometimes called in the older literature a 'universe of discourse'. Quine's use of 'ontology' is roughly equivalent. Here, however, we prefer to speak of the domain (*Bereich*) of individuals of a language-system. This appears to be the terminology most frequently used by mathematical logicians.

It is useful for some purposes to distinguish individuals in a *wider* sense from individuals in a *narrower* sense. Individuals in the narrower sense might be construed as objects of lowest type in a formalism based upon the simplified theory of types, objects of higher type being individuals in the wider sense. (See Chapter VI, §§ C–G). The distinction can be drawn in just this way, however, only for languages based upon simplified type theory. What we desire is a distinction that can be drawn for other kinds of formalisms as well. One might attempt to describe the difference in terms such as 'abstract' or 'concrete', but this would seem to be of little help because these terms themselves are in need of clarification. One very important differentia that is useful is the following. Some domains of individuals are either *finite* or *denumerably infinite*; others are *non-denumerable*.[1] (A domain of individuals of a language may be de-

[1] A *denumerable* or *countable* domain of objects is, roughly speaking, one which can be put into a one-to-one correspondence with the set of natural numbers, and a *non-denumerable* domain is an infinite domain for which no such correspondence exists.

numerable or non-denumerable without our being able to *prove* that this is the case *within* the language itself. Whether this can be proved or not depends in part upon the power of the language at hand. But if a domain of individuals is denumerable, we must at least be able to show or prove that it is within a syntactical or semantical meta-language. See §§ I–J below.) One may then speak of an individual in the *narrow* or *narrower* sense as a member of an at most denumerable domain of individuals; and of individuals in the *wide* or *wider* sense as members of a non-denumerable domain. Accordingly we shall frequently speak of a first-order language in the narrow or narrower sense, meaning thereby a first-order language whose fundamental domain is at most denumerable. And similarly, we can speak of a first-order language in the wide or wider sense as one whose fundamental domain is non-denumerable.

All the languages L are first-order languages with or without identity in either the narrow or wider sense. *Wherever there is choice, a first-order L in the narrow sense is preferable to one in the wider sense.* The reasons for this preference will be urged below in some detail in the discussion of first-order constructivism. (See Chapter XIII.)

It might be thought that the confinement to finite, denumerable, or non-denumerable first-order languages would be too limiting, and that some very important kinds of language-systems would therefore be neglected. It is well known, however, that this is not the case. Most of the important language-systems (of the classical kind) have been formulated *ab initio* as first-order languages, or else can easily be reformulated as such. Languages which would ordinarily be regarded as languages of higher order, such as languages based upon the simplified theory of types or upon the Zermelo set theory, will not be neglected in the present treatment. (See Chapter VI.)

The first-order languages sketched above are such as to contain a single style of variables, but this is by no means essential. If the fundamental domain of a language consists of individuals of various, perhaps mutually exclusive, kinds, it is frequently convenient to use more than one style of variables.[1] Some of the first-order meta-languages we shall be concerned with later contain two distinct styles of variables. Each style is to be regarded as ranging over an appropriate kind of entity. The fundamental domain of the language consists of all of these entities together. In having separate styles of

[1] See A. Schmidt, "Über Deduktive Theorien mit Mehreren Sorten von Grunddingen", *Mathematische Annalen*, 115 (1938), pp. 485–506; and H. Wang, "Logic of Many-Sorted Theories", *The Journal of Symbolic Logic*, 17 (1952), pp. 105–16.

variables, we are aided conceptually in keeping the two kinds of objects distinct. However, it is quite inessential formally to have the two kinds of variables, and one can always reformulate the language so as to contain only a single style if desired. (See (IV,B).)

If L contains two styles of variables a few slight changes must be made in the material above. The definitions of 'formula' and 'term' will be such as to accommodate the two styles of variables. In place of *R5* we need now two rules, one for each style of variable. And similarly for *R6* and *Gen*, for *Abst*, and for *R7* and *R8*. These changes may easily be supplied.

I. Object-Language and Meta-Language. Often the domain of individuals of L is taken as consisting of objects which are non-linguistic in character, and most scientific and philosophical languages are presumably of this kind. But in a second type of discourse, one might wish to speak, not only of non-linguistic objects, but of the very words, phrases, sentences, etc., which comprise that discourse. Or, in a third kind, one might wish to speak *merely* of the expressions of a language and not of the objects to which they refer.

A language which is to be formulated, analyzed, investigated, or described in some way or another within a given context is called an *object-language*. Usually, although not always, the objects about which it speaks are themselves non-linguistic. Languages of the other two kinds mentioned are then called *meta-languages*, more particularly, meta-languages with respect to or of a given object-language.[1] Within a meta-language of L one can talk about the linguistic expressions of L, and perhaps also about the objects of which L itself speaks. Given a meta-language of L, within *its* meta-language one can speak in turn of the linguistic expressions of the meta-language and perhaps also of the objects, linguistic or otherwise, of which it speaks. In this way one gains a whole hierarchy of languages. The lowest language in the hierarchy is ordinarily the object-language. Next comes its meta-language, then its meta-meta-language (i.e., the meta-language of the meta-language), then the meta-meta-meta-language, and so on. Actually there is no upper bound to the hierarchy, because given a language of any level there is always a language of next higher level as its meta-language. In practice we can usually restrict our discourse to languages of two or three levels. Some studies make use of at least four levels, but this is somewhat unusual.

In the preceding sections, §§ A–G, we took L as object-language.

[1] Cf. especially R. Carnap, *The Logical Syntax of Language* (Kegan Paul, London: 1937), pp. 1–9.

In speaking of the signs and expressions of L we used a language closely akin to ordinary English as a meta-language. Actually the meta-language was closely akin to only a very small part of English. Every sentence in §§ A–G could no doubt be translated into Basic English, e.g., in fact into a very small part of Basic English.[1] And on two or three occasions some meta-meta-linguistic sentences were used to say something about the notation of the meta-language. The meta-meta-language was also a language closely akin to a very small fragment of English.

Most of the formalized language-systems which have been studied in detail throughout the literature of modern logic have been taken as object-languages in the sense described. The nature of the meta-language, as in the formulation above of L, is usually left at loose ends. One is not always clear as to precisely what modes of expression the meta-language is allowed to contain. Many logicians use extremely powerful meta-languages containing multifoliate means of expression, languages so powerful in fact as to contain most of mathematics. Such meta-languages have played an important role historically.

But it is also of interest to consider meta-languages of a less powerful kind, in which the modes of expression are extremely limited. In the present book such meta-languages are our primary concern. When one employs a powerful meta-language, containing virtually the whole of mathematics, an explicit formalization of it is not needed. Various ways of formalizing mathematics exist, and one can simply choose a convenient one. But when the meta-language is to be severely limited in means of expression, it is essential to know precisely what concepts, what axioms, modes of inference, etc., are permitted. Here to formalize the meta-language with meticulous care is of the utmost importance.

J. Syntax and Semantics. Two kinds of meta-languages have in effect been distinguished in the preceding section. Given an object-language L, the meta-language may be a language in which we talk exclusively of the linguistic expressions of L; this would be a meta-language of L in the first sense. Or we may have a meta-language of L in the second sense, comprising modes of speaking not only about the linguistic expressions of L but also about the objects of which in turn L speaks. The first kind of a meta-language, in which one speaks exclusively of linguistic expressions, is usually called a *syntax language*, or a *syntactical meta-language* of L, and a language

[1] See C. K. Ogden, *Basic for Science* (Kegan Paul, London: 1942).

of the second kind is called a *semantical meta-language* of L.[1] Of course, a syntactical or semantical meta-language may be rich or weak in modes of expression. It may or may not contain a powerful logic by way of substructure; it may or may not contain powerful axioms or rules of deduction. For any L, there are several different kinds of syntactical and semantical meta-languages of L. And it is to be expected that within a very powerful meta-language of L one can prove statements about L that would not be provable within a restricted meta-language.

This characterization of 'syntax' and 'semantics' is very rough and approximate. A more precise characterization will emerge below from the formulation of specific syntactical and semantical meta-languages.

It has already been remarked that all the language-systems we shall be concerned with in any detail, either as object- or as meta-languages, will be first-order languages. The *object*-languages can be taken in either the narrow or wider sense. The *syntactical meta*-languages to be studied will always be taken in the *narrow* sense. Our aim in part is to show how it is possible to construct *semantical* meta-languages in the *narrow* sense even where the object-languages (in their standard interpretations) are first-order languages in the *wider* sense. (The details are given in Chapters VIII–IX.) As a preliminary to the formulation of semantical meta-languages of this very narrow kind, we shall be concerned with semantical meta-languages which are of the narrow kind if and only if the object-languages under consideration are also (Chapters IV–VII). Although the object-languages considered throughout can be any appropriate systems of first order, we attempt to narrow the semantical meta-languages as much as possible.

K. Sign-Designs and Sign-Events. Another distinction that will prove to be important in the sequel is that between what Peirce called *types* and *tokens*, or between what are now more frequently called *sign-designs* (or *shapes*) and *sign-events* (or *inscriptions*).[2] Heretofore we have spoken of the expressions of L but have not in any precise way specified what these expressions are supposed to be. Also we have spoken of the *symbols* of a language L. The symbols

[1] This usage is essentially that of Tarski and Carnap. Cf. A. Tarski, *op. cit.*, and R. Carnap, *Introduction to Semantics* (Harvard University Press, Cambridge: 1942), pp. 8–15.

[2] Cf. C. S. Peirce, *Collected Papers* (Harvard University Press, Cambridge: 1931–1935), especially vol. 2, §§ 245–6, and vol. 4, § 537; and R. Carnap, *Introduction to Semantics*, pp. 4–8.

are presumably the minimal units or elements or letters of a language, and the expressions are then finite sequences of them. But both 'symbol' and 'expression' are ambiguous. Let us consider first 'symbol'. A symbol may be a particular occurrence in space-time, or it may be a class of such occurrences all sufficiently similar to each other. In the former case we call the symbol a *symbol-event*; in the latter case, a *symbol-design*. Symbol-events are presumably actual occurrences of some kind within the spatio-temporal world. They are singular and particular, occupy a certain portion of space-time, and are no more. A symbol-design, on the other hand, is an entity of an altogether different kind. It is an abstract entity in the sense of having instances. E.g., the symbol-design 'a' has as an instance a particular, concrete, perhaps visible, chalk- or ink-mark in some specific spatio-temporal region. Symbol-designs are presumably best construed as classes of symbol-events. But of course not just *any* class of symbol-events is to be thought of as comprising a symbol-design. As already suggested, the members of a symbol-design must all be sufficiently similar to one another, of similar shapes, sizes, etc. (An exact characterization of what is meant here by 'sufficiently similar' is not required for present purposes.)

Likewise an *expression* may be construed as a sequence of symbol-events taken in a certain order, in which case expressions then also occupy space-time, are particular, etc. Or expressions may be construed as abstract entities having instances. We could distinguish these two meanings of 'expression' by speaking of expression-events and expression-designs. However, there is no need to multiply our vocabulary in this way, and we shall speak of symbol-events or expression-events as *sign-events*, and of expression- or symbol-designs as *sign-designs*.

There are then two quite distinct views concerning the nature of signs or expressions. The view that signs should be construed as sign-designs is called the *classical* view. This view is classical in two senses: (1) syntax and semantics were first formulated on the basis of this view, and (2), the view construes expressions as *classes* of objects. The view that signs should be construed as sign-events is the *inscriptional* or *nominalist* view. This view appears to have originated with Leśniewski, but has received a systematic treatment only recently in the hands of Goodman and Quine. Of course, one can also take a mixed view, regarding signs for some purposes as the one, for other purposes, as the other. In a semantics or syntax of this mixed kind, one would presumably have variables ranging over sign-events as well as class variables ranging over sign-designs. (Cf. Chapter XII, § G.)

In this book syntax and semantics will first be formulated in accord with the classical view (III–X). But later (XI–XII) we shall show how syntax and semantics can be nominalized without substantial loss of power.

L. Some First-Order Object-Languages. Before going on to our main task, the study in detail of certain syntactical and semantical first-order meta-languages, let us glance briefly at some well-known language-systems which can be taken as object-languages. We choose only a few such languages by way of examples. Each language is an *L* with appropriate non-logical rules.

The first examples are two systems closely akin to systems formulated by Hilbert and Bernays for portions of elementary number theory. (They are of especial interest in connection with the proof of the consistency of elementary number theory.) The first system (A) contains the following primitives.[1]

Logical Constants: 'v', '$\sim$', '(', ')', '$\supset$', '='.
Predicate Constant: '$<$'.
Individual Constant: '0'.
Functional Constant: 'σ'.
Variables: 'x', 'x'', 'x''', . . .

The '$<$' is to be thought of as standing for the relation of *being less than* (among natural numbers), '0' for the natural number *zero*, and 'σ' for the function of being the *successor of* a natural number (so that σx is the successor of x). The variables range over natural numbers. The definitions of 'formula' and 'term' can be given informally as in (II,A). Other logical constants are definable by *D1–D4* as in (II,A). The *Logical Rules* are provided by *R1–R8* and *Abst.* The *Non-Logical Rules* are as follows.

A1. $\vdash x < x' \,.\, x' < x'' :\supset: x < x''$.
A2. $\vdash \sim x = x' :\supset: x < x' \,.\mathrm{v}.\, x' < x$.
A3. $\vdash x < \sigma x$.
A4. $\vdash x < x' \,.\supset.\, \sim x' < \sigma x$.
A5. $\vdash \sim x < 0$.
A6. $\vdash \sim x = 0 \,.\supset.\, (\mathrm{E}x')\sigma x' = x$.

The Rules of Inference are *MP* and *Gen.* (Note here, incidentally, that *R7* is provable by *A2–A4* and hence can be dropped as one of the logical rules.)

The second example is in essential respects the famous Hilbert-Bernays system (Z).[2] The logical constants here are as in (A). There

[1] Cf. D. Hilbert and P. Bernays, *op. cit.*, vol. 1, p. 263.
[2] *Ibid.*, p. 371.

are no primitive non-logical predicate constants. 'o' is the only primitive individual constant, and 'σ', '+', and '$\times$' are the primitive functional constants. The variables 'x', 'x''', etc., range over natural numbers as in (A), 'o' designates the natural number zero, and 'σ', '+', and '$\times$' stand respectively for the successor, addition, and multiplication functions of elementary arithmetic. The rules comprise, in addition to logical rules analogous to those of (A), the following.

Z1. $\vdash \sim\sigma x = \mathrm{o}$.
Z2. $\vdash \sigma x = \sigma x' . \supset . x = x'$.
Z3. $\vdash x + \mathrm{o} = x$.
Z4. $\vdash x + \sigma x' = \sigma(x+x')$.
Z5. $\vdash x \times \mathrm{o} = \mathrm{o}$.
Z6. $\vdash x \times \sigma x' = (x \times x') + x$.
Z7. $\vdash F\mathrm{o} . (x)(Fx . \supset . F\sigma x) : \supset : Fx$.

Z1, *Z2*, and *Z7* are forms of *Peano's Postulates* and *Z3–Z6* are the so-called *recursion equations* for addition and multiplication. *Z7* is a principle of *mathematical induction*. The mathematical theory characterized by these axioms is generally called *elementary number theory*.

Several kinds of first-order languages have been studied recently for methodological purposes by Carnap. The systems $\mathfrak{L}_1, \mathfrak{L}_2, \ldots, \mathfrak{L}_\infty$, discussed in *Logical Foundations of Probability*, e.g., are closely akin to first-order languages describable roughly as follows.[1] The primitives include the logical constants (as above), a finite number of primitive predicate constants each of specified finite degree (such as 'Blue', 'Warmer', etc.), a finite or denumerable number of individual constants, together with variables. The logic presupposed is essentially that provided by *R1–R8*, *MP*, *Gen*, and *Abst*, to which appropriate non-logical axioms are added. Each $\mathfrak{L}_n$ is a system whose fundamental domain contains just n individuals, whereas $\mathfrak{L}_\infty$ contains a denumerable number of individuals.

Another variety of first-order system studied by Carnap is the coordinate language $\mathfrak{L}'$.[1] The individuals of such a language are assumed to possess a fixed basic order, akin to that of the natural numbers arranged in order of increasing magnitude. The individuals may be thought of as *positions* in a coordinate system. Also it may be desirable to have the arithmetic of natural numbers available in such a system. If so, two different types of primitive predicate constants are needed. The first express properties of numbers as such;

[1] R. Carnap, *Logical Foundations of Probability*, pp. 52–70; and *Meaning and Necessity*, p. 75 and pp. 78–81.

the second, properties of numbers as positions. E.g., if 'Prim' is a primitive predicate constant standing for the property of being a prime number, 'Prim 3' may state that the number 3 is a prime number. Where 'Blue' is another primitive predicate constant, 'Blue 3' on the other hand may state that the position whose coordinate is 3 is blue. The underlying logic of such a language is essentially as above. The non-logical axioms required include statements stipulating the basic order of the individuals, if this is not already provided by a suitable choice of the primitive notation.

In each of the languages $\mathfrak{L}_n$, $\mathfrak{L}_\infty$, or $\mathfrak{L}'$ certain areas of scientific theory are to be presumed formalizable. The primitives must be chosen in such a way as to assure the definability of basic terms and concepts of the science at hand. In spite of the very narrow underlying logical framework, language-systems of this kind can be surprisingly powerful in modes of expression.

The systems of Hilbert-Bernays and Carnap mentioned are all first-order languages in the narrower sense, in which the fundamental domain of individuals is taken to be at most denumerable. Each is intended to enable us to formalize certain areas of *science*. An interesting example of a *philosophical* first-order language-system, also in the narrow sense, is provided by the constructional system of Goodman.[1] This may be described very roughly as follows. By a *quale* one means, following C. I. Lewis, a presented character or quality as over against a property or "logical universal".[2] Goodman construes 'quale' in a broad sense so as to include places and times but not sizes and shapes. The fundamental domain of objects for Goodman consists of qualia together with entities which have at least one quale as a part. The basic individuals are, in other words, qualia together with sums of them. The underlying first-order logic is supplemented by a calculus of individuals, concerned with notions such as the overlapping of individuals, the part-whole relation between individuals, sums and products of individuals, etc. This calculus can be formulated in such a way as to presuppose no logical notions other than those of a first-order language. The primitive logical constants may be taken as above. (Strictly, identity is not required as a primitive logical constant, being definable in terms of the other primitives.) The primitive predicate constants include four two-place relation symbols. Goodman shows how three of these may be replaced by one. A full specification of the meaning of the primitive predicate constants requires considerable preliminary explana-

[1] See N. Goodman, *The Structure of Appearance*.

[2] See C. I. Lewis, *Mind and the World Order* (Scribner's, New York: 1929), especially pp. 60–1.

tion and need not be given here. The predicate constants take qualia and sums of them as arguments. Goodman does not consider except in passing the problem of giving suitable non-logical axioms characterizing the notions embodied in these primitives. His system is thus only partially formalized. But presumably consistent axioms can be given of sufficient deductive power to provide for the theory Goodman has in mind.

Several language-systems of interest primarily for philosophy or methodology have been constructed which, in their normal interpretation, are not of first order, e.g., Carnap's *L*II,[1] the system of Carnap's *Logische Aufbau der Welt*, Woodger's system of axiomatic biology,[2] the Hull–Fitch system for the theory of rote learning,[3] etc. Practically all of these systems, however, allow themselves to be reformulated as first-order systems, in which the domains of individuals include classes, relations, etc. Systems of this kind will be discussed in Chapter VI.

[1] R. Carnap, *The Logical Syntax of Language*, pp. 83–151.

[2] J. H. Woodger, *The Axiomatic Method in Biology*, and *The Technique of Theory Construction* (*International Encyclopedia of Unified Science*, vol. II, no. 5: 1939).

[3] C. Hull, F. B. Fitch, etc., *op. cit.*

CHAPTER III

FIRST-ORDER SYNTAX

LET us now formulate in detail a syntactical meta-language of a quite restricted kind. It will be a first-order language in the narrow sense, in which the fundamental domain consists of the expressions (sign-designs) of L, where L is the object-language. Within this narrow syntactical meta-language, we shall be able to give a full syntactical description of L, i.e., we shall be able to define the basic concepts of syntax relativized to L. A syntax of this kind is called a *first-order* or *elementary syntax*, the logical concepts and methods used being of the most elementary kind.

The syntax to be developed here is of the classical kind in which the expressions of L are taken as sign-designs. We assume explicitly an infinity of expressions of L. Every primitive expression of L is assumed to exist in the appropriate sense; and given any two expressions of L, no matter how long, the expression consisting of the first followed immediately by the second is also an expression of L. Each expression is of finite length, but the totality of all expressions of L is denumerably infinite. This procedure is consonant with assumptions that are frequently made in physical geometry, physics, and mathematics, and is in some respects the most convenient way of arranging the matter. (For a discussion of finitism, infinitism, nominalism, and related topics, see Chapter XIII.)

Already in the preceding chapter, we have formulated L somewhat roughly within an informal meta-language. Some of the basic concepts of syntax relativized to L are there already defined informally. E.g., the notion of being a primitive logical constant of L is defined by enumeration. Also the notion of being an axiom is there defined as any expression of such and such a form. The definition of 'formula' is, however, of the recursive kind, as is also the definition of 'theorem'.

The aim of the present chapter is to make all these definitions fully explicit by incorporating them within a formal system. We shall note precisely the primitive notions being presupposed, the assumptions being made, and the means of logical deduction being employed. The requirement that the logical means be restricted to those of a narrow first-order language entails that we relinquish some of the powerful tools that are ordinarily employed in syntax.

E.g., the convenience of recursive definition is closed to us, the theory of recursions requiring for its adequate formalization logical notions and assumptions of a powerful kind. (See (VII,E).) The recursive definitions given informally in the preceding chapter will not be formalizable here at all. Instead we shall have to seek some quite different means of defining 'formula of L' and 'theorem of L'. The gaining of satisfactory definitions of these notions will constitute one of the main technical problems of this chapter. The method employed will be an adaptation of the methods of Chwistek, Tarski, and Quine.[1]

Let the formalized meta-language here be called M. The object-language is any one of the languages L with identity in either the narrow or wider sense. To simplify procedure somewhat, let us assume here and throughout that L contains at least one primitive predicate constant, but *no functors and no individual constants as primitive* unless otherwise specified. This entails no serious limitation, as we have seen in (II,F), and will enable us to simplify some definitions. But M itself is also a first-order language of the kind discussed in Chapter II, and hence is another example of an L with identity. It is, however, a first-order language in the narrow sense, its fundamental domain of objects being denumerable. Being a *syntactical* meta-language, within it we speak exclusively of the linguistic expressions of L. M will contain both individual constants and a functor as primitives.

Let us proceed immediately to a rigorous formulation of M. In § A the *primitives* of M will be presented, and the *rules* governing them given in § B. In § C some *elementary theorems* of M will be proved. Some preliminary *syntactical definitions* will be given in § D, leading up to the *definition of 'Axiom of L'* in § E, and to that of *'Theorem of L'* in § F. Finally, in § G some *further syntactical theorems* will be given, and in § H a few additional *syntactical notions* will be defined.

A. Primitives. The logical primitives of the meta-language M will be precisely those of L in Chapter II. As variables of the meta-language, let us use

$$\text{'}a\text{'}, \text{'}b\text{'}, \text{'}c\text{'}, \text{'}d\text{'}, \text{'}e\text{'}, \text{'}a'\text{'}, \text{'}b'\text{'}, \ldots, \text{'}a'''\text{'}, \ldots.$$

The use of primes or accents in this way enables us to form a new variable wherever needed. Occasionally we may form new variables also by means of numerical subscripts. The 'v' and '$\sim$' are to have their usual meaning, and '$=$' and '$\supset$' are to be handled as above.

[1] L. Chwistek, *The Limits of Science* (Kegan Paul, London: 1948), pp. 83–100 and 162–91; and W. V. Quine, *Mathematical Logic*, pp. 291–305. Also A. Tarski, *op. cit.*, pp. 291–303.

But in addition M is to contain several individual constants and one important functor (or functional constant). Recall now that the primitive signs of the object-language L are

'v', '$\sim$', '(', ')', '=', '$\ni$', 'x', 'x''', . . ., 'P', 'P$''$', . . ., and 'P$''''$'.

Now 'x''', or 'P$''''$', e.g., need not be regarded as indivisible expressions. 'x''' consists of an 'x' followed by ' ' '; 'x'''' of 'x' followed by ' ' ' followed by ' ' '; and so on. So that we can really think of the primitive signs of L as being just nine in number, taking ' ' ' as a primitive.

For explicitness when talking about the sign '(' of L, it will be helpful to have a specific sign of M. Let '*lp*' be a proper name of '('. (*lp* may then be thought of, if desired, as the class of the left-parenthetical sign-events of L.) Similarly, let us assume that

'*rp*'	names ')'	(right parenthesis),
'*vee*'	names 'v',	
'*tilde*'	names '$\sim$',	
'*id*'	names '='	(identity sign),
'*invep*'	names '$\ni$'	(inverted epsilon),
'*ex*'	names 'x',	
'*pee*'	names 'P',	

and

'*ac*'	names ' ' '	(accent).

'*lp*', '*rp*', . . ., '*ac*' are then primitive individual constants of M, and each of them is the name of a corresponding primitive symbol of L.

Within M, we shall not introduce quotation marks. We achieve clarity and avoid the sometimes confusing use of quotation marks if we make systematic use of '*vee*', '*tilde*', etc., instead. (See, however, (XII,F).)

In addition to these nine primitive individual constants, there is also need in M for a functor designating a certain operation or function. This will be symbolized by '$\frown$'. The arguments of this function are the individuals of M and its functional values likewise are the individuals of M. The operation or function for which this functor stands may be called *concatenation*, and $(a \frown b)$ may be called the *concatenate* of the expressions a and b. Given a and b, the concatenate $(a \frown b)$ is the expression consisting of *a followed immediately by b*. E.g., $(lp \frown ex \frown rp)$ is simply the expression '(x)' and $(lp \frown ex \frown rp \frown ex \frown id \frown ex)$[1] is the expression '$(x)x{=}x$'. The operation of concatenation is obviously essential in M; without it we should have no way of forming

[1] Some unneeded inner parentheses are omitted here. This is justified by *TC2* below.

names of the expressions of L which contain more than one primitive symbol. It enables us to make explicit the operation involved whenever we write one letter of a language followed immediately by another. This important operation was first studied by Gödel and Tarski.

When we speak of one expression a of L being "followed immediately" by another b we are of course speaking somewhat abstractly, because a and b are sign-*designs*. When one speaks of one sign-*event* being followed by another, assuming that both sign-events be visible (or audible or otherwise perceptible), one can give a quite definite, "concrete" meaning to concatenation. Under these circumstances the concatenate of two expressions is the sign-event consisting of the first perceptibly followed by the other, somewhat in the way in which one word-event on a printed line is followed by another. But where the arguments are sign-designs, no such concrete interpretation is available. Sign-designs are in some sense "abstract" entities, and the result of applying the operation of concatenation to two sign-designs is itself a sign-design, and hence abstract in the same sense. Of course one could require that corresponding to each sign-design there exist at least one sign-event. The corresponding sign-event would presumably be a *member* of the given sign-design. Given then what it means to say that one sign-event is followed by another, we could explain what it means to say that one sign-design is followed by another. The sign-design $(a \frown b)$ would consist of just those sign-events which in turn consist of a member of a followed by a member of b. But for the present, we are concerned exclusively with sign-designs. Sign-events are in no way values for variables in M, and we do not explicitly assume that there is a sign-event corresponding to each sign-design. Hence the somewhat "abstract" character to the meaning of '$\frown$'. (Cf. Chapters XI and XII below.)

It should be emphasized that M is a very limited language-system. Within it we can speak only of the concatenation and identity of expressions of L. No other modes of expression are available, other than those permitted by the underlying first-order logic.

The expression used in M to name a corresponding expression of L is called the *structural description* of that expression of L. Note that the notion of a structural description is not definable within M. It is definable only in a meta-language in which we can speak of the symbols of M as well as those of L, i.e., within a comparative meta-language of M and L. '*lp*' is the structural description of '(', '*rp*', of ')', etc. And the result of writing a primitive individual constant of M followed by '$\frown$' followed by a primitive individual constant of M,

etc., is the structural description of the corresponding expression of L. To use again an earlier example, '$(lp \frown ex \frown rp \frown ex \frown id \frown ex)$' is the structural description of '$(x)x = x$'. The process of giving the structural description of an expression of L is a kind of logical spelling. One spells out successively the constituent primitive symbols of the expression, much as one spells out successively the constituent letters of a word.

Note that within M, we cannot form any of the sentences of L. All we can do within M is to talk *about* the expressions of L. Within M we *mention* the expressions of L, we do not *use* them. Strictly speaking, even the primitive logical constants of L should be regarded as symbols distinct from the corresponding primitive logical constants of M. However, the role which these constants play within L and M is so similar that it seems desirable to use the same symbols in both languages. Within M, it might be thought that we both use and mention, e.g., 'v'. We use it to form a disjunction within M, as in

$$'(lp = tilde \text{ .v. } \sim lp = tilde)',$$

whereas we mention it in

$$'\sim vee = tilde'.$$

It might therefore be thought that in

$$'(vee = tilde \text{ .v. } \sim vee = tilde)'$$

we are both mentioning and using 'v'. Strictly speaking, however, '*vee*' is taken as the structural description of the 'v' of L only.

B. Rules. Let us now consider the axiomatic structure of M. The axioms fall conveniently into two groups, the logical axioms and the specifically syntactical ones.

The *Logical Axioms* are provided by *R1–R8* and *Abst* as above, the rules of inference being *MP* and *Gen*.

The *Axioms of Syntax* are provided by the following three rules.[1] The first stipulates that no two of the primitive symbols of L are identical, i.e., that *vee* is not identical with *tilde*, with *lp*, with *rp*, or etc. Let us write '$a \neq b$' as is customary in place of '$\sim a = b$'. The first rule, a *Rule of Distinctness*, reads as follows.

[1] These are due essentially to Tarski. See *Der Wahrheitsbegriff*, p. 289 and p. 383. Tarski's other axioms are automatically provided for here. With regard to *SynR4*, see also R. Carnap, *The Logical Syntax of Language*, p. 38 and *passim*.

(*Note added in proof:* An English translation of *Der Wahrheitsbegriff* has recently appeared. See Tarski's *Logic, Semantics, Metamathematics*, tr. by J. H. Woodger (Oxford University Press, Oxford: 1956).)

SynR1. $\vdash vee \neq tilde \,.\, vee \neq lp \,.\, vee \neq rp \,.\, vee \neq id \,.\, vee \neq invep \,.\, vee \neq ex \,.\, vee \neq ac \,.\, vee \neq pee \,.\, tilde \neq lp \,.\, \ldots \,.\, ac \neq pee.$

The second rule, also a *Rule of Distinctness*, states that no primitive symbol of L is identical with any concatenate of expressions of L.

SynR2. $\vdash \sim(\mathrm{E}a)(\mathrm{E}b)vee = (a \frown b) \,.\, \sim(\mathrm{E}a)(\mathrm{E}b)tilde = (a \frown b) \,.\, \ldots \,.\, \sim(\mathrm{E}a)(\mathrm{E}b)ac = (a \frown b).$

The next rule is a *Rule of Identity* stating the circumstances under which two concatenates of expressions of L are to be regarded as identical. The concatenate of a and b is identical with that of c and d, if and only if either (i) a is identical with c and b with d, or (ii) b is the concatenate of some e with d where c is the concatenate of a with that e, or (iii) a is the concatenate of c with some e and d is the concatenate of that e with b.

SynR3. $\vdash (a \frown b) = (c \frown d) \,.\equiv.\, ((a = c \,.\, b = d) \,.\mathrm{v}.\, (\mathrm{E}e)(b = (e \frown d) \,.\, c = (a \frown e)) \,.\mathrm{v}.\, (\mathrm{E}e)(a = (c \frown e) \,.\, d = (e \frown b))).$

The fourth and last rule required is a syntactical rule of inference, a *Rule of Infinite Induction*. It tells us that from an infinite set of formulae, each member of which states that a certain property holds of a given (fixed) symbol or (fixed) concatenate of symbols of L, one can infer that this property holds of all expressions. Let '$\boldsymbol{F}$' stand for any one-place abstract of M, as explained in the preceding chapter. The rule may then read as follows.

SynR4. If every formula which results from '$\boldsymbol{F}a$' by putting in a structural-descriptive expression of M in place of 'a' is a theorem, then $\vdash (b)\boldsymbol{F}b$.

If a given property holds of lp, rp, etc., of $(lp \frown rp)$, $(lp \frown ex \frown ac \frown ac)$, etc., etc. (exhausting all possible structural-descriptive phrases of M), then we can conclude by this rule that the given property holds of all expressions of L. Note it is essential here to allow for an *infinite* number of formulae in the hypothesis, one for each expression of L. This rule stipulates in effect that the nameable expressions of L, i.e., the expressions for which there are structural descriptions in M, exhaust the expressions of L.

It might be thought that in admitting within M an infinitistic rule of this kind, we are in some sense transgressing the narrow, first-order bounds to syntax and semantics that have been laid down. But such rules have in no way been forbidden and the admission of them is quite in accord with what is meant here by a denumerable, first-order system. Of course, one is free to reject infinitistic rules of this kind if one wishes. But their admission is not only very convenient; it seems essential for the formulation of some of the

semantical meta-languages below. (But no such rule will be required for non-translational semantics, as we shall see later.) And here we assume that the object-language L contains no such rule. The notion of a *proof* for a language M containing such a rule must be somewhat broader than that informally sketched in the preceding chapter, because any use of this rule involves an infinity of steps. Proofs ordinarily involve only a finite number of steps, as we have already noted. The notion of *proof in M*, nonetheless, will be clear from examples. (Cf. (V,H), (IX,D), and (XIII,E).)

C. Some Elementary Properties of Concatenation. Let us consider now a few very elementary theorems concerning concatenation. The various laws concerning truth-functions, quantifiers, abstraction, and identity (of Chapter II) are presupposed.

First we note that the concatenate of any two expressions *exists*.

TC1a. $\vdash (a)(b)(\mathrm{E}c)c = (a \frown b)$.

The proof is immediate using *R7*. Also, in view of identity theory, we have immediately a *uniqueness* law, that

TC1b. $\vdash (a)(b)(c)(d)(a = (b \frown c) \,.\, d = (b \frown c) :\supset: a = d)$.

Using *TC1a* we can prove the law of *associativity*.

TC2. $\vdash (a \frown (b \frown c)) = ((a \frown b) \frown c)$.

By *SynR3*, we know that

$$\vdash e = (b \frown c) \,.\, e' = (a \frown b) :\supset: (a \frown e) = (e' \frown c).$$

But from this we can get *TC2*, using *TC1a* (to drop the hypotheses) and some identity laws.

By the Rule of Infinite Induction, we can prove another induction law as follows. This law states that if it is a theorem that (i) every primitive sign of L has a certain property and that (ii) for any expressions a and b of L, if a and b both have the property then $(a \frown b)$ has it also, then it is a theorem that every expression of L whatsoever has the property. Let

'PS a' abbreviate '(a = *vee* .v. a = *tilde* .v. a = *lp* .v. a = *rp* .v. a = *id* .v. a = *invep* .v. a = *pee* .v. a = *ex* .v. a = *ac*)'.

'PS a' can read 'a is a primitive symbol-design of L'. The new inductive principle is then as follows.

TC3a. If $\vdash (a)(\mathrm{PS}\ a \,.\supset.\, \boldsymbol{F}a) \,.\, (a)(b)\ (\boldsymbol{F}a \,.\, \boldsymbol{F}b :\supset: \boldsymbol{F}(a \frown b))$, then $\vdash (a)\boldsymbol{F}a$.

The proof is not difficult. Let 'Hyps' stand for

$$\text{'}(a)(\mathrm{PS}\ a \,.\supset.\, \boldsymbol{F}a) \,.\, (a)(b)(\boldsymbol{F}a \,.\, \boldsymbol{F}b :\supset: \boldsymbol{F}(a \frown b))\text{'}.$$

First we note that, by quantification theory,

$$\vdash \text{Hyps} :\supset: \boldsymbol{F}\ vee,$$
$$\vdash \text{Hyps} :\supset: \boldsymbol{F}\ tilde,$$

etc. But also,

$$\vdash \text{Hyps} : \supset: \boldsymbol{F}(vee \frown vee),$$
$$\vdash \text{Hyps} :\supset: \boldsymbol{F}(vee \frown tilde),$$

and so on for all possible concatenates. Hence we can show that if

$$\vdash \text{Hyps},$$

then every sentence which results from '$\boldsymbol{F}a$' by putting in a structural-description in place of 'a' is a theorem. But if this latter holds we know, by *SynR4*, that then

$$\vdash (a)\boldsymbol{F}a.$$

Hence, if

$$\vdash \text{Hyps},$$

then

$$\vdash (a)\boldsymbol{F}a.$$

But this is *TC3a*. (A slightly stronger induction rule will be proved in § G below.)

By using this induction law we can show that if a concatenate $(a \frown b)$ is identical with a concatenate $(a \frown c)$ then b is identical with c.

TC3b. $\vdash (a \frown b) = (a \frown c) .\supset. b = c.$

The proof is by induction according to *TC3a*. By *SynR3* we know that

$$\vdash \text{PS}\ a\ .\ (a \frown b) = (a \frown c) :\supset: b = c\ .\text{v}.\ (\text{E}e)(b = (e \frown c)\ .\ a = (a \frown e))\ .\text{v}.\ (\text{E}e)(a = (a \frown e)\ .\ c = (e \frown b)).$$

But by *SynR2* and some identity laws,

$$\vdash \text{PS}\ a .\supset. \sim(\text{E}e)a = (a \frown e).$$

Hence

$$\vdash \text{PS}\ a\ .\supset.\ (b)(c)((a \frown b) = (a \frown c)\ .\supset.\ b = c).$$

But also,

$$\vdash (b)(c)((a \frown b) = (a \frown c)\ .\supset.\ b = c)\ .\ (b)(c)((d \frown b) = (d \frown c)\ .\supset.\ b = c)$$
$$: \supset: (b)(c)(((a \frown d) \frown b) = ((a \frown d) \frown c)\ .\supset.\ b = c),$$

using *TC2* and some identity laws. Hence *TC3b* by *TC3a*.

SynR2 tells us that none of the primitive symbols of L is a concatenate. Some *expressions* of L are clearly concatenates, of course. But no expression a is a concatenate of a with anything or of anything with a.

TC4a. $\vdash a \neq (a \frown b).$
TC4b. $\vdash a \neq (b \frown a).$

The proofs are by induction. By *SynR2* and some identity laws,

(1) $\vdash \mathrm{PS}\ a\ .\supset.\ (c)a \neq (a \frown c).$

But also we have that

(2) $\vdash (c)a \neq (a \frown c)\ .\ (c)b \neq (b \frown c) :\supset: (c)(a \frown b) \neq ((a \frown b) \frown c).$

To see this we note that by *SynR3*,

$\vdash (a \frown b) = ((a \frown b) \frown c) :\supset: (a = (a \frown b)\ .\ b = c)\ .\mathrm{v}.\ (\mathrm{E}e)(b = (e \frown c)\ .\ (a \frown b) = (a \frown e))\ .\mathrm{v}.\ (\mathrm{E}e)(a = ((a \frown b) \frown e)\ .\ c = (e \frown b)).$

But

$$\vdash a = (a \frown b)\ .\ b = c :\supset: a = (a \frown c),$$
$$\vdash b = (e \frown c)\ .\ (a \frown b) = (a \frown e) :\supset: b = (b \frown c),$$

using *TC3b*, and

$$\vdash a = ((a \frown b) \frown e)\ .\supset.\ (a \frown b) = (((a \frown b) \frown e) \frown b),$$

and hence, by *TC2* twice and *TC3b*,

$$\vdash a = ((a \frown b) \frown c)\ .\ c = (e \frown b) :\supset: b = (b \frown c).$$

From these we easily get (2). Using (1) and (2) we get *TC4a* by *TC3a*.

The proof of *TC4b* is similar. By *TC4a* or *TC4b* we also have immediately that

TC4c. $\vdash a \neq (a \frown a).$

Concatenation is not commutative.

TC5. $\vdash \sim(a)(b)(a \frown b) = (b \frown a).$

We see this at once if we note, by *SynR1*, that

$$\vdash \sim lp = rp.$$

Hence, by *SynR3* and *SynR2*,

$$\vdash \sim(lp \frown rp) = (rp \frown lp).$$

Hence,

$$\vdash (\mathrm{E}a)(\mathrm{E}b)\sim(a \frown b) = (b \frown a),$$

from which we get *TC5*.

An *expression* of L is any primitive symbol of L or any concatenate.

'Exp a' abbreviates '(PS a .v. (Eb)(Ec)$a = (b \frown c)$)'.

One of the most important uses of the induction rule *TC3a* is to enable us to prove that all the objects which are values for variables in M are expressions of L.

TC6. $\vdash (a)\mathrm{Exp}\ a.$

Note that

$$\vdash \mathrm{PS}\ a.\supset.\ \mathrm{Exp}\ a,$$

and

$$\vdash \mathrm{Exp}\ b\ .\ \mathrm{Exp}\ c\ .\ a = (b \frown c) :\supset: \mathrm{Exp}\ a.$$

Hence *TC6*, by *TC3a*.

Also in virtue of *SynR2*, *TC6*, and the definitions,

TC7. $\vdash \sim\mathrm{PS}\ a \mathrel{.\equiv.} (\mathrm{E}b)(\mathrm{E}c)a = (b \frown c)$.

D. Some Preliminary Definitions. Several basic concepts of syntax (relativized to L) will now be defined within M.[1]

An expression a of L is said to *begin* an expression b if b consists of a concatenated with some c. A more useful relation than this is that of *beginning or of being identical with*. Let 'Bgn' symbolize this relation. The definition is as follows.

'a Bgn b' abbreviates '$((\mathrm{E}c)b = (a \frown c) \mathrel{.v.} a = b)$'.[2]

E.g., '$(x)x$' begins the expression '$(x)x = x$' in the sense of this definition. In a similar way one defines the relation of *ending or of being identical with*.

'a Ends b' abbreviates '$((\mathrm{E}c)b = (c \frown a) \mathrel{.v.} a = b)$'.

E.g., '$= x$' ends '$(x)x = x$'.

In terms of 'Bgn' and 'Ends' one can define the notion of being a *segment* or *consecutive part of*.

'a Seg b' abbreviates '$(\mathrm{E}c)(a$ Ends $c \mathrel{.} c$ Bgn $b)$'.

a is a segment or consecutive part of b if and only if a ends some c which begins b. The expression 'v Px'', e.g., is a consecutive part of the formula '$(x)(\sim\mathrm{P}x \vee \mathrm{P}x')$'. Note that the definition does not allow disconnected or scattered segments of an expression.

An expression a is a *string of accents* if it consists of ac concatenated with itself any finite number of times. Of course the notion of a string of accents cannot be defined in this way within M, without a prior definition of 'finite number of times'. But one can say that a is a string of accents if and only if every primitive symbol contained in a is ac.

'AcString a' abbreviates '$(b)(\mathrm{PS}\ b \mathrel{.} b$ Seg $a :\supset: b = ac)$'.

Any expression of L such as '''''' . . .' ' is an AcString in this sense.

An expression a of L is a *variable* if and only if a is ex or consists of ex followed by one or more accents.

[1] These definitions are for the most part adaptations of those of Quine, *op. cit.*, pp. 295–305.

[2] This definition is to be construed in the general sense of containing any distinct variables of M in place of 'a' and 'b', the variable 'c' being then chosen as some variable distinct from 'a' and 'b'. Similar remarks are to apply to the definitions given in the preceding section and throughout. Definitions containing variables of L are to have the effect throughout of what are sometimes called *definitional schemata*.

'Vbl a' abbreviates '($a = ex$ .v. (Eb)(AcString b . $a = (ex \frown b)$))'.

It was stipulated above that the variables of L are to be taken as 'x', 'x''', 'x'''', etc. The reason for this stipulation was simply to facilitate here the definition of 'Vbl'. If variables such as 'y', 'z', etc., were also admitted in L, this definition would perforce be more complicated.

In a similar way one might define the notion of being a *primitive predicate constant*. An expression a might be called such if and only if it is either *pee* or *pee* followed by a string of accents. We do not need so general a definition as this, however, because there are only a specified finite number of primitive predicate constants in L, and each of these is of a specified degree. Let us suppose that we have only five primitive predicate constants, as in (II,A), and that 'P' is of degree 1, 'P'' is of degree 2, . . ., and 'P''''' is of degree 5.

We need then here only the following more restricted definition.

'PrimPredCon a' abbreviates '($a = pee$ .v. $a = (pee \frown ac)$ .v. $a = (pee \frown ac \frown ac)$ .v. $a = (pee \frown ac \frown ac \frown ac)$ .v. $a = (pee \frown ac \frown ac \frown ac \frown ac)$)'.

The notion of an *atomic* formula of L may now be defined as a formula consisting of '=' with variables on either side or of a primitive predicate constant of degree n ($1 \leq n \leq 5$) followed by n variables.

'AtFmla a' abbreviates '(Eb)(Ec)(Ed)(Ee)(Ea')(Vbl b . Vbl c . Vbl d . Vbl e . Vbl a' : $a = (b \frown id \frown c)$ .v. $a = (pee \frown b)$ .v. $a = (pee \frown ac \frown b \frown c)$. v. $a = (pee \frown ac \frown ac \frown b \frown c \frown d)$ .v. $a = (pee \frown ac \frown ac \frown ac \frown b \frown c \frown d \frown e)$ .v. $a = (pee \frown ac \frown ac \frown ac \frown ac \frown b \frown c \frown d \frown e \frown a')$)'.

The definition of 'AtFmla' for an L containing more (or fewer) than five primitive predicate constants is similar. (Recall that L contains no individual or functional constants. If it did, this would have to be taken into account in this definition. See (X,A).)

Before defining the notion of *formula* in full generality, we need some further preliminary definitions.

First we wish to define what it means to say that an expression a is a result of quantifying b universally, or that a results from b by prefixture of a universal quantifier to b. To say this is merely to say that there is some variable c such that a consists of the following expressions in order: left parenthesis, c, right parenthesis, b.

'a Gen b' abbreviates '(Ec)(Vbl c . $a = (lp \frown c \frown rp \frown b)$)'.

The left-hand side of this definition may be read 'a is a *generalization* of b'. Note that here b can be any expression of L at all, not just a formula, although in the important uses of this definition, b will in fact be a formula.

The notion embodied in '*a* Gen *b*' corresponds in a certain way with the *Rule of Generalization* as stated informally in the preceding chapter. Where *a* and *b* are formulae, '*a* Gen *b*' says that *a* is obtainable from *b* by means of *Gen*. The next definition defines a notion corresponding in a similar way to the *Rule of Modus Ponens*.

'*a* MP *b,c*' abbreviates '(c = (lp⌢*tilde*⌢b⌢*vee*⌢a⌢rp) .v. b = (lp⌢*tilde*⌢c⌢*vee*⌢a⌢rp))'.

To say '*a* MP *b,c*' amounts essentially, in the case where *a*, *b*, and *c* are formulae, to saying that *a* is obtainable from *b* and *c* by means of the rule *MP*. Of course 'MP' is defined here for all expressions *a*, *b*, and *c* as arguments, although the definition will be useful only where *a*, *b*, and *c* are formulae.

An expression *a* is an *immediate consequence* of *b* and *c* if and only if *a* can be reached from *b* and *c* by means of *MP* or from *b* by means of *Gen*.

'*a* IC *b,c*' abbreviates '(*a* MP *b,c* .v. *a* Gen *b*)'.

An expression *a* is obtainable from *b* by *abstraction* if and only if there are variables *c* and *d* such that *a* is (c⌢*invep*⌢b⌢d).

'*a* Abst *b*' abbreviates '(E*c*)(E*d*)(Vbl *c* . Vbl *d* . *a* = (c⌢*invep*⌢b⌢d))'.

Because we shall frequently have occasion to speak of concatenates such as (lp⌢a⌢*vee*⌢b⌢rp), (*tilde*⌢a), (lp⌢*tilde*⌢a⌢*vee*⌢b⌢rp), etc., the following definitions are useful.

'(*a vee b*)' abbreviates '(lp⌢a⌢*vee*⌢b⌢rp)',
'*tilde a*' abbreviates '(*tilde*⌢a)',

and

'(*a hrsh b*)' abbreviates '(*tilde a vee b*)'.

This last definition introduces '(*a hrsh b*)' as the structural description of the expression of *L* consisting of *a* followed by '⊃' followed by *b*, the whole expression enclosed in parentheses. But this is merely an abbreviation for (lp⌢*tilde*⌢a⌢*vee*⌢b⌢rp). (Cf. *D*1 of (II,A).) Similarly let

'(*a dot b*)' abbreviate '*tilde* (*tilde a vee tilde b*)',

and

'(*a tripbar b*)' abbreviate '((*a hrsh b*) *dot* (*b hrsh a*))'.

(Cf. *D*2 and *D*3 of (II,A).) Also let

'*a qu b*' abbreviate '(lp⌢a⌢rp⌢b)',

and

'*a exisqu b*' abbreviate '(*tilde a qu tilde b*)'.

For appropriate a and b, these two definitions introduce notations respectively for universal and existential quantifiers in context.

We use '*vee*' rather than '*Vee*', in order to remind ourselves that '(*a vee b*)' is a *term* of M, not a formula. And similarly with '*tilde*', '*dot*', etc. Symbols for predicates are capitalized throughout, and hence if we wrote '(*a Vee b*)', this might be mistaken for a formula of M.

The notion of being a formula will be defined by Quine's method of *framed ingredients*. Recall that, intuitively speaking, a formula is an expression which can be obtained from atomic formulae by means of a finite number of applications of negation, disjunction, quantification, or abstraction. Given any formula, one should then be able to separate out its constituent formulae as proper parts. E.g., speaking informally, a disjunction '$(A \vee B)$' is separable into A and B, a negation '$\sim A$' into A, and a quantification '$(x)A$' into A. Given any formula, one can continue this kind of separation into parts until atomic formulae are reached as the ultimate constituents. E.g., consider the formula

$$'(x)(A_1 \vee A_2) \supset ((y)(B_1 \vee \sim B_2) \,.\, (\mathrm{E}y)(A_1 \supset B_2))',$$

where A_1, A_2, B_1, and B_2 are atomic formulae and x and y are variables. Written wholly in primitive notation, this becomes

(1) $'(\sim(x)(A_1 \vee A_2) \vee \sim(\sim(y)(B_1 \vee \sim B_2) \vee \sim\sim(y)\sim(\sim A_1 \vee B_2)))'$.

The left disjunct of this may be decomposed successively into

(*a*) $'\sim(x)(A_1 \vee A_2)'$, $'(x)(A_1 \vee A_2)'$, $'(A_1 \vee A_2)'$, A_1, A_2.

Similarly the right-hand disjunct of (1) may be decomposed successively into

(*b*) $'\sim(\sim(y)(B_1 \vee \sim B_2) \vee \sim\sim(y)\sim(\sim A_1 \vee B_2))'$, $'\sim(y)(B_1 \vee \sim B_2) \vee \sim\sim(y)\sim(\sim A_1 \vee B_2)'$, $'\sim(y)(B_1 \vee \sim B_2)'$, $'\sim\sim(y)\sim(\sim A_1 \vee B_2)'$, $'(y)(B_1 \vee \sim B_2)'$, $'\sim(y)\sim(\sim A_1 \vee B_2)'$, $'B_1 \vee \sim B_2'$, $'(y)\sim(\sim A_1 \vee B_2)'$, B_1, $'\sim B_2'$, $'\sim(\sim A_1 \vee B_2)'$, B_2, $'\sim A_1 \vee B_2'$, $'\sim A_1'$, B_2, A_1.

We could now form the following sequence of formulae from the list (*a*),

(2) A_1, A_2, $'(A_1 \vee A_2)'$, $'(x)(A_1 \vee A_2)'$, $'\sim(x)(A_1 \vee A_2)'$,

and a similar sequence for the right-hand disjunct from the list (*b*),

(3) B_1, B_2, A_1, $'\sim B_2'$, $'(B_1 \vee \sim B_2)'$, $'\sim A_1'$, $'(\sim A_1 \vee B_2)'$, . . .

Let us then form the sequence consisting of the members of the sequence (2), followed by those of (3) deleting repetitions, and finally add the formula (1) as the last member of the sequence. Call this resulting sequence (4).

It seems evident that a formula could be regarded as the sort of

expression which can be built up in this way from *prior* members of a sequence of the kind (4) by means of quantification, abstraction, disjunction, and negation. We might try to mirror this idea in a formal definition. We should have no problem if we had already defined 'sequence' and 'prior'. But sequences are ordinarily regarded as one-many relations, the converse domains of which consist of the natural numbers.[1] Priority might then be construed as the relation between numbers of being less than. But because M lacks relational as well as numerical variables, these notions are not available. Rather than sequences, we might employ long expressions formed by concatenating in order the expressions which would otherwise be members of the sequence. In place of the sequence (4) we should have an expression consisting of A_1 concatenated with A_2 concatenated with '$(A_1 \vee A_2)$' . . . concatenated finally with (1). The difficulty with taking such expressions in place of sequences is that we should not be able to speak of the *parts* of such an expression in the same way that we are able to speak of the *members* of a sequence. A_1 concatenated with A_2 would be a part of the long expression, e.g., but there would be no member of the sequence (4) corresponding to this. Hence we must search for a method of speaking of the members of the long expressions and not just of parts indiscriminately. The suggestion is to frame on either side the constituents by some arbitrary string of symbols which could not significantly occur within any constituent. Each constituent would then be a *framed ingredient* or a framed constituent of the long expression. To perform the framing one can use any arbitrary expression not otherwise a part of a formula, e.g., '( )', i.e., $(lp \frown rp)$. Now let us mirror this method, described informally, within formal definitions.

An expression a is said to be a *framed ingredient* of b if and only if the result of framing a on either side by means of '( )' is a consecutive part of b, and '( )' is itself not a consecutive part of a.

'a FrIng b' abbreviates '$((lp \frown rp \frown a \frown lp \frown rp)$ Seg $b. \sim(lp \frown rp)$ Seg $a)$'. Where $(lp \frown rp)$ does not occur in a', b', . . ., c', and b is

$$(lp \frown rp \frown a' \frown lp \frown rp \frown b' \frown lp \frown rp \frown \ldots \frown lp \frown rp \frown c' \frown lp \frown rp),$$

each of a', b', . . ., c' is a framed ingredient of b. Note that any other string of symbols of L which could not significantly occur within a formula could be used equally well here in place of $(lp \frown rp)$.

[1] Recall that those objects which stand in a given dyadic relation R to some object constitute the *domain* of that relation, and those objects to which something bears R constitute its *converse domain*. (Also an object which is in the domain or the converse domain of R is in the *field* of R.) And a dyadic relation R is *one-many* if and only if for all x, y, and z in its field, if x bears R to z and also y bears R to z then x and y are identical.

A relation of *priority* is definable as follows.

'a Pr_b c' abbreviates '(Ed)(d Bgn b . a FrIng d . c FrIng b . $\sim c$ FrIng d)'.

An expression a is prior to c in b if and only if there is an expression d which begins b and of which a is a framed ingredient, and c is a framed ingredient of b but not of d. Intuitively the notion of priority involved here seems obvious enough.

A formula is now definable as a framed ingredient of some expression b, every framed ingredient of which is either an atomic formula, or is the negative of a prior framed ingredient of b, or is the disjunction of two prior ingredients of b, or is the result of quantifying universally a prior formula of b, or is gained from a prior formula by abstraction.

'Fmla a' abbreviates '(Eb)(a FrIng b . (c)(c FrIng b :⊃: AtFmla c .v. (Ed)(Ee)(d Pr_b c . e Pr_b c : c = (*tilde* d) .v. c = (d *vee* e) .v. c Gen d .v. c Abst d)))'.

This definition gives a fully general definition within M of the notion of being a *formula of L*.

Note that by means of framed ingredients the effect of a recursive definition is achieved. The prior-most ingredients of b are the atomic formulae, the next (reading from left to right) are the simplest molecular formulae, then formulae of greater complexity, and so on. The device of using framed ingredients will enable us to define several notions below also which are otherwise definable recursively. (See (VII,E).) Note also that this definition makes precise the notion of formula sketched informally in (II,A), where, however, L is taken as perhaps containing one or more primitive individual or functional constants.

The formulae as construed here comprise both sentences and proper sentential functions. In order to differentiate between these the notion of being a *free occurrence of a variable* in a given expression must be defined. There has been no need up to this point to distinguish between an expression and an *occurrence* of an expression. But when we speak of free variables it is essential to make the distinction. The first occurrence of 'x' is *free* in

(5) '(Px v (x)Px)',

e.g., whereas the second and third are *bound*. How can we distinguish between different occurrences of one and the same variable within a given context? One way of construing occurrences is in terms of beginning segments. Subject to a refinement, let us say that a is an *occurrence* of an expression b in c if and only if a begins c and b ends a.

To construe occurrences in this way might seem *prima facie* somewhat artificial. But it enables us to distinguish different occurrences of the same symbol or expression, and strictly this is all that is required. Within (5) '(Px' is the first occurrence of the variable 'x', '(Px v (x' is the second, and '(Px v (x)Px' is the third. The definition of the notion of occurrence is as follows.

'a Occ$_c$ b' abbreviates '(a Bgn c . b Ends a . $\sim(a \frown ac)$ Bgn c)'.

The added clause '$\sim(a \frown ac)$ Bgn c' is desirable here, because we should not wish to regard occurrences of initial (beginning) segments of variables or primitive predicate constants as occurrences of variables or primitive predicate constants respectively. It would seem unnatural to regard the variable 'x''', e.g., as having an occurrence within

(6) 'Px'''''.

Yet without the added clause 'Px''' would be an occurrence of the variable 'x''' in (6). (See the end of § E below for an alternative method which does not employ the notion of occurrence.)

We may also wish to say that *a occurs* in *b*. This is to say merely that there is a c which is an occurrence of a in b.

'a Occ b' abbreviates '(Ec)c Occ$_b$ a'.

To say that an occurrence of a variable a in b is a *bound* occurrence of that variable in b, as we have already seen in Chapter II (§§ A and D), is to say that that occurrence is in a part of b of the form (a *qu* e) or ($a \frown invep \frown e$) where e is a formula. More precisely, within M we can say that an occurrence a of a variable b in c is a bound occurrence of b in c if and only if there is an expression a' which is a formula and either there is a b' and a c' such that a is ($b' \frown b$) and (i) ($b' \frown c'$) is an occurrence of (b *qu* a') in c and (b *qu* a') does not end c' or (ii) ($b' \frown c'$) is an occurrence of ($b \frown invep \frown a'$) in c and ($b \frown invep \frown a'$) does not end c'; or ($a \frown invep \frown a'$) is a segment of c.

'a BOcc$_c$ b' abbreviates '(Vbl b . a Occ$_c$ b . (Ea')(Fmla a' : (Eb')(Ec')(a = ($b' \frown b$) : (($b' \frown c'$) Occ$_c$ (b *qu* a') . $\sim$(b *qu* a') Ends c') .v. (($b' \frown c'$) Occ$_c$ ($b \frown invep \frown a'$) . $\sim$($b \frown invep \frown a'$) Ends c')) .v. ($a \frown invep \frown a'$) Seg c))'.

It is not difficult to verify that this definition is intuitively adequate.

To say that an occurrence c of a variable a in b is a *free* occurrence is to say in effect that it is not bound.

'c FrOcc$_b$ a' abbreviates '(c Occ$_b$ a . Vbl a . $\sim c$ BOcc$_b$ a)'.

Occasionally we may wish to say that a variable a is a *free variable* of b without specifying the occurrence.

'a FV b' abbreviates '(Ec)c FrOcc$_b$ a'.

The notions of being a proper sentential function and of being a sentence are now definable. An expression *a* is said to be a *proper sentential function* (or *open formula* of *L*) if it is a formula and contains at least one free variable.

'PrSentFunc *a*' abbreviates '(Fmla *a* . (E*b*)*b* FV *a*)'.

And an expression *a* is a *sentence* (or a *closed formula*) if it is a formula and contains no free variables.

'Sent *a*' abbreviates '(Fmla *a* . ~(E*b*)*b* FV *a*)'.

Some elementary theorems concerning these various notions will be given in § G below.

E. 'Axiom of L'. The notion of being an *axiom* of *L* may now be defined rigorously with *M*.

The axioms of *L* stipulated informally in the preceding chapter by *R1* are expressions of the form

'(*A* v *A*) ⊃ *A*'.

Hence we can let

'TFAx1 *a*' abbreviate '(E*b*)(Fmla *b* . *a* = ((*b vee b*) *hrsh b*))'.

An expression *a* of *L* is a truth-functional axiom of the first kind if and only if there is a formula *b* such that *a* is the expression ((*b vee b*) *hrsh b*).

Similarly

'TFAx2 *a*' abbreviates '(E*b*)(E*c*)(Fmla *b* . Fmla *c* . *a* = (*b hrsh* (*b vee c*)))',

'TFAx3 *a*' abbreviates '(E*b*)(E*c*)(Fmla *b* . Fmla *c* . *a* = ((*b vee c*) *hrsh* (*c vee b*)))',

and

'TFAx4 *a*' abbreviates '(E*b*)(E*c*)(E*d*)(Fmla *b* . Fmla *c* . Fmla *d* . *a* = ((*b hrsh c*) *hrsh* ((*d vee b*) *hrsh* (*d vee c*))))'.

The correspondence between these three definitions and the rules *R2*, *R3*, and *R4*, respectively, is similar to the correspondence between *R1* and the definition of 'TFAx1 *a*'.

Recall now that *R5* states, where *L* contains no primitive terms other than variables, that

$$\vdash (x)A \supset B,$$

if ***B*** differs from ***A*** only in containing free occurrences of some one variable ***y*** wherever there are free occurrences of the variable ***x*** in ***A***. In order to define the notion of being an axiom of quantification of this kind, a rigorous definition of the phrase '***B*** differs from ***A*** only

(etc.)' is needed. Rather than to define this directly, we can define it in three steps as follows.

First let 'a $\mathrm{SF1}^b_c$ d' symbolize that a differs from d only in containing a free occurrence of b in place of just *one* free occurrence of c in d. The 'S' and 'F' are to suggest 'substitution' and 'free'. One obtains a by substituting b for a free occurrence of c in d (provided the resulting occurrence of b is free in a). This can be defined by requiring that there are expressions a' and b' such that a' is a free occurrence of c in d, b' is a free occurrence of b in a, and for all e, a' is $(e \frown c)$ if and only if b' is $(e \frown b)$, and for all e, d is $(a' \frown e)$ if and only if a is $(b' \frown e)$.

'a $\mathrm{SF1}^b_c$ d' abbreviates '$(\mathrm{E}a')(\mathrm{E}b')(a'\ \mathrm{FrOcc}_d\ c\ .\ b'\ \mathrm{FrOcc}_a\ b\ .\ (e)(a' = (e \frown c)\ .\equiv.\ b' = (e \frown b))\ .\ (e)(d = (a' \frown e)\ .\equiv.\ a = (b' \frown e)))$'.

That this definition gives what we wish may be seen by considering an example. For, if a is

$$\text{'}(\mathrm{P}x' \vee (x)\mathrm{P}x)\text{'}$$

and d is

$$\text{'}(\mathrm{P}x \vee (x)\mathrm{P}x)\text{'},$$

where c is 'x' and b is 'x''', then 'a $\mathrm{SF1}^b_c$ d' holds.

This definition may be extended to the case where we consider 0 or *more than one* free occurrences of c in d, as follows.

'a $\mathrm{SF?}^b_c$ d' abbreviates '$(\mathrm{E}e)(a\ \mathrm{FrIng}\ e\ .\ (a')(a'\ \mathrm{FrIng}\ e :\supset: a' = d.\ \mathrm{v}.\ (\mathrm{E}b')(b'\ \mathrm{Pr}_e\ a'\ .\ a'\ \mathrm{SF1}^b_c\ b')))$'.

'a $\mathrm{SF?}^b_c$ d' holds then provided there exists an expression e of which a is a framed ingredient and every framed ingredient a' of e is either d or there is a prior ingredient b' of e such that a' $\mathrm{SF1}^b_c$ b'.

It is now easy to define the notion that a differs from d only in containing free occurrences of b *wherever* there are free occurrences of c in d.

$$\text{'}a\ \mathrm{SF}^b_c\ d\text{'} \text{ abbreviates } \text{'}(a\ \mathrm{SF?}^b_c\ d : c\ \mathrm{FV}\ a\ .\supset.\ b = c)\text{'}.$$

'a SF^b_c d' holds just where 'a $\mathrm{SF?}^b_c$ d' holds provided no free occurrences of c are left in a (unless of course b and c are the same variable).

The definition of being an axiom of quantification stipulated by *R5* or *R6* may now be given. An expression a is a quantificational axiom of the first kind if and only if there are a formula e and expressions b, c, and d, such that b SF^c_d e and a is $(d\ qu\ e\ hrsh\ b)$.

'QuantAx1 a' abbreviates '$(\mathrm{E}b)(\mathrm{E}c)(\mathrm{E}d)(\mathrm{E}e)(\mathrm{Fmla}\ e\ .\ b\ \mathrm{SF}^c_d\ e\ .\ a = (d\ qu\ e\ hrsh\ b))$'.

Similarly

'QuantAx2 a' abbreviates '$(\mathrm{E}b)(\mathrm{E}c)(\mathrm{E}d)(\mathrm{Vbl}\ b\ .\ \mathrm{Fmla}\ c\ .\ \mathrm{Fmla}\ d\ .\ \sim b\ \mathrm{FV}\ c\ .\ a = (b\ qu\ (c\ vee\ d)\ hrsh\ (c\ vee\ b\ qu\ d)))$'.

The notion of being an axiom of identity is also readily definable. To be an axiom of identity stipulated by *R7*, is to be of the form $(a \frown id \frown a)$ where a is a variable.

'IdAx1 a' abbreviates '$(\mathrm{E}b)(\mathrm{Vbl}\ b\ .\ a = (b \frown id \frown b))$'.

And to be an identity axiom of the kind stipulated by *R8* may be defined as follows.

'IdAx2 a' abbreviates '$(\mathrm{E}b)(\mathrm{E}c)(\mathrm{E}d)(\mathrm{E}e)(\mathrm{Fmla}\ e\ .\ \mathrm{Vbl}\ b\ .\ \mathrm{Vbl}\ c\ .\ \mathrm{Vbl}\ d\ .\ a = (b \frown id \frown c\ hrsh\ (d \frown invep \frown e \frown b\ hrsh\ d \frown invep \frown e \frown c)))$'.

Having included the Axioms of Abstraction among the logical axioms of L, we must also define the notion of being such an axiom.

'AbstAx a' abbreviates '$(\mathrm{E}b)(\mathrm{E}c)(\mathrm{E}d)(\mathrm{E}e)(\mathrm{Fmla}\ e\ .\ d\ \mathrm{SF}?^{b}_{c}\ e\ .\ \sim b\ \mathrm{FV}\ e\ .\ a = (b \frown invep \frown d \frown c\ tripbar\ e))$'.

Finally an expression a is a *logical axiom* of L if and only if it is an axiom of one of the nine kinds defined.

'LogAx a' abbreviates '(TFAx1 a .v. TFAx2 a .v. TFAx3 a .v. TFAx4 a .v. QuantAx1 a .v. QuantAx2 a .v. IdAx1 a .v. IdAx2 a .v. AbstAx a)'.

We specifically call these the *logical* axioms. For, as already remarked, we should ordinarily add to these axioms further ones characterizing the properties or relations (of L) P, P′, etc. Some specific non-logical axioms for various L were given at the end of the preceding chapter. We shall not consider any of these axioms in further detail at present. But it is clear enough from the foregoing how one could define rigorously within M the notion of being a descriptive or non-logical axiom of L. Let 'DesAx a' express that a is such an axiom. To be an *axiom* of L is then to be either a descriptive or a logical axiom of L.

'Ax a' abbreviates '(LogAx a .v. DesAx a)'.

Later, in Chapter VI, we shall consider the non-logical axioms of certain especially important kinds of object-languages.

F. 'Theorem of L'. A *theorem* is the sort of formula one gains by using the rules of Modus Ponens and Generalization a finite number of times starting with the axioms. And a *logical* theorem is a theorem gained in this way by starting with logical axioms. In terms of framed ingredients, one can say that an expression a is a logical theorem of L if and only if it is a framed ingredient of an expression b every framed ingredient of which is either itself a logical axiom or is an immediate consequence of prior framed ingredients of b.

'LogThm a' abbreviates '(Eb)(a FrIng b . (c)(c FrIng b :⊃: LogAx c .v. (Ed)(Ee)(d Pr_b c . e Pr_b c . c IC d,e)))'.

Similarly, to say that a is a theorem, symbolized 'Thm a', can be defined by replacing 'LogAx c' in the definiens here by 'Ax c'.

The notion of theorem is one of the most important of (elementary) syntax. Upon its basis any other syntactical ideas which may be needed are definable. Hence we have in the restricted language M an adequate basis for the syntax of L.

It may also be desirable occasionally to speak of a *descriptive theorem*, defined as follows.

'DesThm a' abbreviates '(Thm a . ~LogThm a)'.

Two or three further notions which will be useful in the sequel are definable as follows.

An expression a is said to be a *sentential function of* (*at most*) *the one variable* b if and only if a is a formula, b may or may not be a free variable of a, but every free variable of a is the same as b.

'SentFuncOne a,b' abbreviates '(Fmla a . Vbl b . (c)(c FV a .⊃. $c = b$))'.

In a similar way one can say that a is a sentential function of (at most) two variables b and c.

'SentFuncTwo a,b,c' abbreviates '(Fmla a . Vbl b . Vbl c . ~$b = c$. (d)(d FV a : ⊃ : $d = b$ .v. $d = c$))'.

The notion of being a primitive predicate constant has already been defined. We now define a related notion, that of being a *one-place predicate constant*. The one-place predicate constants include the one-place primitive predicate constant 'P' and in addition those (one-place) abstracts which contain no free variables. Any such abstract expresses a property and hence can appropriately be regarded as a (one-place) predicate constant, as we have already noted in (II,D).

'PredCon a' abbreviates '(a = *pee* .v. (Eb)(Ec)(SentFuncOne c,b . $a = (b \frown invep \frown c)$))'.

(The use of 'constant' to cover abstracts containing no free variables may be somewhat unusual, but will be convenient throughout.)

An *atomic universal sentence* of L is any sentence of the form '(x)A' where A contains no free variable other than x, as we have already noted in (II,G).

'AtUnivSent a' abbreviates '(Eb)(Ec)(SentFuncOne c,b . $a = (b$ *qu* $c)$)'.

A *universal sentence* is then any expression one can obtain by a finite number of applications of disjunction or negation starting with

atomic universal sentences. Hence this notion is definable by the method of framed ingredients.

'UnivSent a' abbreviates '(Eb)(a FrIng b . (c)(c FrIng b :⊃: AtUnivSent c .v. (Ed)(Ee)(d Pr_b c . e Pr_b c : c = (d *vee* e) .v. c = (*tilde* d))))'.

Clearly every sentence of L is either an atomic universal sentence or a non-atomic universal sentence. (Cf. *TG9h* below.) The notions of atomic universal sentence and universal sentence will be especially useful in defining the semantical truth-concept for L in (V,A).

The notion of being a *string of universal quantifiers* may be defined as follows.

'QuantString a' abbreviates '(Eb)(a FrIng b . (c)(c FrIng b :⊃: (Ed) (Vbl d . c = ($lp \frown d \frown rp$)) .v. (Ed) (Ee) (d Pr_b c . e Pr_b c . c = ($d \frown e$))))'.

And an expression a which contains no free variables and consists of a string of universal quantifiers prefixed to b is said to be a *closure* of b.

'a Clsr b' may abbreviate '((Ec)(QuantString c . a = ($c \frown b$)) . ~(Ed)d FV a)'.

According to this definition any expression of L may have a closure, but clearly the most important closures will be closures of formulae. Also it is often convenient to speak of closures in such a way that a *sentence* is its *own* closure. For this we let

'a Clsr' b' abbreviate '((Sent b . a = b) .v. (~Sent b . a Clsr b))'.

Perhaps the somewhat artificial notion of *occurrence*, defined in § D above, upon which many of the notions of syntax depend (as we have seen), is in some sense counter-intuitive. At any event, there seems little reason to identify occurrences with beginning segments, as we did there, except for the technical convenience it affords. And we can easily define essentially these same notions, if desired, without using the notion of occurrence. In order to see this, we introduce, by means of framed ingredients, a definition involving *simultaneous induction*, i.e., induction with respect to two variables. (These methods can easily be extended to handle induction with respect to three or more variables also.) As preliminaries we may define

'(a;b) FrIng c' as '(Ed)(d FrIng c . d = ($a \frown vee \frown vee \frown b$) . ~($vee \frown vee$) Seg a . ~($vee \frown vee$) Seg b)',

'(a;b) Pr_e (c;d)' as '(Ea')(a' Bgn e . (a;b) FrIng a' . (c;d) FrIng e . ~(c;d) FrIng a')',

and

'a FV b' as '(Vbl a . (Ec)(b FrIng c . (d)(d FrIng c :⊃: d = ($pee \frown a$) .v. (Ea')(Eb')(Ec')(Ed')(Vbl a' . Vbl b' . Vbl c' . Vbl d' : d = ($pee \frown$

$ac \frown a \frown a')$ .v. $d = (pee \frown ac \frown a' \frown a)$ .v. $d = (pee \frown ac \frown ac \frown a \frown b' \frown c')$ .v. $d = (pee \frown ac \frown ac \frown b' \frown a \frown c')$ .v. ... .v. $d = (pee \frown ac \frown ac \frown ac \frown ac \frown a' \frown b' \frown c' \frown d' \frown a))$ .v. (Ea') (Eb')(Vbl a' . Fmla b' . $d = (a' \frown invep \frown b' \frown a))$ .v. (Ee)(e Pr_c d : $d = (tilde\ e)$ .v. (Ea')(Fmla a' : $d = (a'$ *vee* $e)$ .v. $d = (e$ *vee* $a'))$ .v. (Eb')(Vbl b' . $\sim a = b'$. $d = (b'$ *qu* $e))$ .v. (Eb')(Ec')(Vbl b'. Vbl c' . $\sim a = b'$. $\sim a = c'$. $d = (b' \frown invep \frown e \frown c')))))'$.

Then by a kind of simultaneous induction,

'a SF?^b_c d' abbreviates '(Vbl b . Vbl c : ($\sim c$ FV d . $a = d$) .v. (Ee) ($(a;d)$ FrIng e . $(a')(b')((a';b')$ FrIng e :$\supset$: $(b' = (pee \frown c)$. $(a' = (pee \frown b)$ .v. $a' = b'))$ .v. (Ea'') (Eb'') (Ec'') (Ed'') (Vbl a'' . Vbl b'' . Vbl c'' . Vbl d'' : $(b' = (pee \frown ac \frown c \frown a'')$. $(a' = (pee \frown ac \frown b \frown a'')$ .v. $a' = b'))$ .v. $(b' = (pee \frown ac \frown a'' \frown c)$. $(a' = (pee \frown ac \frown a'' \frown b)$ .v. $a' = b'))$ .v. ... .v. $(b' = (pee \frown ac \frown ac \frown ac \frown ac \frown a'' \frown b'' \frown c'' \frown d'' \frown c)$. $(a' = (pee \frown ac \frown ac \frown ac \frown ac \frown a'' \frown b'' \frown c'' \frown d'' \frown b)$ .v. $a' = b'))$ .v. (Ec') (Ed') (Vbl c' . Fmla d' . $b' = (c' \frown invep \frown d' \frown c)$. $(a' = (c' \frown invep \frown d' \frown b)$ .v. $a' = b'))$ .v. (Ec')(Ed')(Ea'')(Eb'')($(c';d')$ Pr_e $(a';b')$. $(a'';b'')$ Pr_e $(a';b')$: $(a' = (tilde\ c')$. $b' = (tilde\ d'))$ .v. $(a' = (c'$ *vee* $a'')$. $b' = (d'$ *vee* $b''))$ .v. (Ee')(Vbl e' . $\sim e' = b$. $\sim e' = c$. $a' = (e'$ *qu* $c')$. $b' = (e'$ *qu* $d'))$ .v. (Ec'')(Ed'')(Vbl c'' . Vbl d'' . $\sim c'' = b$. $\sim c'' = c$: $(a' = (c'' \frown invep \frown c' \frown d'')$. $b' = (c'' \frown invep \frown d' \frown d''))$ .v. $(a' = (c'' \frown invep \frown c' \frown b)$. $b' = (c'' \frown invep \frown d' \frown c)))))))'$.

We shall scarcely use these definitions subsequently. But they are intuitively adequate and provide an alternative way of handling the notion of being a free variable and the notion embodied 'SF?' without using the notion of occurrence. The other syntactical notions needed may then be defined as above.

G. Some Syntactical Theorems. Some elementary theorems of M have been proved above in § C. Some further theorems will be given here. Most of these either have some intrinsic interest in the sense that they enunciate important properties of concatenation or of the various notions defined, or will be useful in later chapters. The proofs for the most part are omitted.

The first few theorems give some obvious properties concerning Bgn, Ends, and Seg.

TG1a. $\vdash a$ Bgn a . a Bgn $(a \frown b)$.
TG1b. $\vdash a$ Bgn b . b Bgn c :$\supset$: a Bgn c.
TG1c. $\vdash a$ Ends a . a Ends $(b \frown a)$.
TG1d. $\vdash a$ Ends b . b Ends c :$\supset$: a Ends c.
TG1e. $\vdash a$ Ends b .$\supset$. a Ends $(c \frown b)$.
TG1f. $\vdash a$ Bgn b .$\supset$. $(c \frown a)$ Bgn $(c \frown b)$.
TG2a. $\vdash a$ Seg a . a Seg $(a \frown b)$. a Seg $(b \frown a)$. a Seg $((b \frown a) \frown c)$.

The proofs of these theorems are immediate. Some of them lead up to the transitivity law for Seg.

TG2b. $\vdash a \text{ Seg } b \,.\, b \text{ Seg } c :\supset: a \text{ Seg } c$.

The proof is obvious if it is broken up as follows. First, by *TG1b*,

$\vdash a \text{ Ends } d \,.\, d \text{ Bgn } b \,.\, b = e \,.\, e \text{ Bgn } c :\supset: (\text{E}d)(a \text{ Ends } d \,.\, d \text{ Bgn } c)$.

But also, using *TG1e*, *TG1f*, and *TG1b*,

$\vdash a \text{ Ends } d \,.\, d \text{ Bgn } b \,.\, e = (a' \frown b) \,.\, e \text{ Bgn } c :\supset: a \text{ Ends } (a' \frown d) \,.\, (a' \frown d) \text{ Bgn } c$.

Combining these appropriately, we get *TG2b*.

By the definition of 'Seg', it will be recalled that a Seg b if and only if a ends some expression which begins b. The following theorem is of interest in showing that this is equivalent to saying that *a begins* some expression which *ends b*.

TG2c. $\vdash a \text{ Seg } b \,.\equiv.\, (\text{E}c)(a \text{ Bgn } c \,.\, c \text{ Ends } b)$.

The proof is immediate in view of the foregoing theorems.

The following laws give evident properties of priority

TG3a. $\vdash a \text{ Pr}_b\, c :\supset: \sim c \text{ Pr}_b\, a \,.\, \sim b \text{ Pr}_c\, a \,.\, \sim a \text{ Pr}_c\, b \,.\, \sim c \text{ Pr}_a\, b \,.\, \sim b \text{ Pr}_a\, c$.

TG3b. $\vdash \sim a \text{ Pr}_a\, a \,.\, \sim a \text{ Pr}_b\, a \,.\, \sim b \text{ Pr}_a\, a \,.\, \sim a \text{ Pr}_a\, b$.

TG3c. $\vdash a = (lp \frown rp \frown b \frown lp \frown rp \frown c \frown lp \frown rp) \,.\, \sim(lp \frown rp) \text{ Seg } b \,.\, \sim (lp \frown rp) \text{ Seg } c :\supset: b \text{ Pr}_a\, c$.

TG3d. $\vdash \sim a = b \,.\, a \text{ FrIng } c \,.\, b \text{ FrIng } c :\supset: a \text{ Pr}_c\, b \,.\text{v}.\, b \text{ Pr}_c\, a$.

We now turn to a very important law, which is useful in proving appropriate properties of notions defined by means of framed ingredients. This law, the *Rule of Framed Ingredients*, reads as follows.

TG4. $\vdash a \text{ FrIng } b \,.\, (c)(c \text{ FrIng } b :\supset: \boldsymbol{F}c \,.\text{v}.\, (\text{E}d)(\text{E}e)(d \text{ Pr}_b\, c \,.\, e \text{ Pr}_b\, c \,.\, \boldsymbol{A})) \,.\, (c)(\boldsymbol{F}c \,.\supset.\, \boldsymbol{G}c) \,.\, (c)(d)(e)(\boldsymbol{G}d \,.\, \boldsymbol{G}e \,.\, \boldsymbol{A} :\supset: \boldsymbol{G}c) :\supset: \boldsymbol{G}a$, if $\boldsymbol{A}$ is a formula of M containing only 'c', 'd', and 'e' as free variables.

Although this rule looks to be long and somewhat complicated, actually it is extremely simple. It states, roughly speaking, that if every framed ingredient of an expression b has a certain property or can be reached from prior ingredients of b by a certain operation (expressed by $\boldsymbol{A}$), and if every expression which has this one property has a second property which is closed under the given operation, then every framed ingredient of b has this second property. Actually the rule is a little more complicated than this, but in only inessential respects. It enables us to achieve the effect of a kind of mathematical induction. The proof is not difficult in view of the Rule of Infinite Induction, and may be sketched roughly as follows. First, take 'b' as the structural description of an expression of L containing no framed

ingredient at all. Then the hypothesis of *TG4*, '*a* FrIng *b*', fails to hold. Suppose now '*b*' is taken as the structural description of an expression containing just one framed ingredient. If *a* is not that framed ingredient, the hypothesis again fails to hold and *TG4* follows. On the other hand, if *a* is that framed ingredient, then there is no prior ingredient, and hence (by the second hypothesis of *TG4*) *a* has the property indicated by ***F***. Hence also, *a* has the property indicated by ***G*** (by the third hypothesis). Therefore *TG4* holds for '*b*' taken as the structural description of an expression of *L* containing just one framed ingredient. If '*b*' is taken as the structural description of an expression containing two framed ingredients identical to each other, the argument is similar. Next take '*b*' as the structural description of an expression containing two distinct framed ingredients. One of these framed ingredients will be prior to the other, by *TG3d*. If *a* is the prior-most ingredient, then by the second and third hypotheses of *TG4*, *a* also has the property indicated by ***G***. If *a* is not the prior-most, we know from the preceding sentence that the prior-most ingredient, say *a'*, has the property indicated by ***G***. By the second hypothesis *a* either has the property indicated by ***F*** or can be reached from ingredients prior to *a* by a certain operation. In the case under consideration *a'* is the only framed ingredient prior to *a* from which *a* can be reached by the operation, and *a'* has the property indicated by ***G***. (By the fourth hypothesis of *TG4* the property indicated by ***G*** holds of expressions reached by performing this operation upon expressions which have this property.) And therefore *a* also has the property indicated by ***G***. If now we take '*b*' as the structural description of an expression containing three or more framed ingredients, the argument is similar. One concludes that *TG4* holds for '*b*' taken as any structural description, and hence, by *SynR4*, for variable '*b*'.

We have also the following lemmas.

TG5a. $\vdash$ Fmla a .$\supset$. Fmla (*tilde* a).

The proof is immediate if we note that

$\vdash a$ FrIng b . $(c)(c$ FrIng $b :\supset:$ AtFmla c .v. $(\mathrm{E}d)(\mathrm{E}e)(d\ \mathrm{Pr}_b\ c\ .\ e\ \mathrm{Pr}_b\ c$: $c = (\mathit{tilde}\ d)$ .v. $c = (d\ \mathit{vee}\ e)$ .v. c Gen d .v. c Abst $d)) .\ a' = (b \frown \mathit{lp} \frown \mathit{rp} \frown \mathit{tilde}\ a \frown \mathit{lp} \frown \mathit{rp}) :\supset: (\mathit{tilde}\ a)$ FrIng a' . $a\ \mathrm{Pr}_{a'}\ (\mathit{tilde}\ a)$. (c) $(c$ FrIng $a' :\supset:$ AtFmla c .v. $(\mathrm{E}d)(\mathrm{E}e)(d\ \mathrm{Pr}_{a'}\ c\ .\ e\ \mathrm{Pr}_{a'}\ c : c = (\mathit{tilde}\ d)$ .v. $c = (d\ \mathit{vee}\ e)$ .v. c Gen d .v. c Abst $d))$.

In a similar way one can prove that

TG5b. $\vdash$ Fmla a . Fmla $b :\supset:$ Fmla $(a\ \mathit{vee}\ b)$,

TG5c. $\vdash$ Fmla a . b Gen $a :\supset:$ Fmla b,

and that

TG5d. $\vdash$ Fmla a . b Abst a :$\supset$: Fmla b.

Using *TG5a–TG5d*, one can show that

TG5e. $\vdash$ TFAx1 a .$\supset$. Fmla a,
TG5f. $\vdash$ TFAx2 a .$\supset$. Fmla a,

and similarly for the other axioms also. These theorems may be summarized by noting that

TG6a. $\vdash$ Ax a .$\supset$. Fmla a.

Also note that

TG6b. $\vdash$ Fmla a . Fmla b . c MP a,b :$\supset$: Fmla c.

Now, using the Rule of Framed Ingredients, one can prove that

TG6c. $\vdash$ Thm a .$\supset$. Fmla a.

By *TG4*,

$\vdash$ a FrIng b . (c)(c FrIng b :$\supset$: Ax c .v. $(\mathrm{E}d)(\mathrm{E}e)$($d$ Pr_b c . e Pr_b c . (c Gen d .v. c MP d,e))) . (c)(Ax c .$\supset$. Fmla c) . $(c)(d)(e)$(Fmla d . Fmla e . (c Gen d .v. c MP d,e) :$\supset$: Fmla c) :$\supset$: Fmla a.

The third hypothesis here may be dropped by *TG6a*, and the fourth by *TG6b* and *TG5c*.

Also we have the following formal statement of the Rule of Generalization.

TG7a. $\vdash$ Thm a . b Gen a :$\supset$: Thm b.

The proof is rather similar to that of *TG5c*. The formal statement of the Rule of Modus Ponens is that

TG7b. $\vdash$ Thm a . Thm b . c MP a,b :$\supset$: Thm c.

The proof is similar to that of *TG6b*.

Several lemmas which may be useful later are listed as follows.

TG8a. $\vdash$ PS a . $\sim a$ = *pee* :$\supset$: $\sim$PredCon a.
TG8b. $\vdash$ SentFuncOne a,b :$\equiv$: PredCon ($b \frown$ *invep* $\frown a$) . Vbl b.
TG8c. $\vdash$ Sent a . Vbl b :$\supset$: SentFuncOne a,b.
TG8d. $\vdash$ Sent a . Sent (a *vee* b) :$\supset$: Sent b.
TG8e. $\vdash$ Sent a .$\supset$. Sent (*tilde* a).
TG8f. $\vdash$ SentFuncOne a,b . Vbl c :$\supset$: $(\mathrm{E}d)d$ SF^c_b a.
TG8g. $\vdash$ PredCon ($a \frown$ *invep* $\frown b$) . Vbl a . $\sim a$ FV b :$\supset$: Sent b.
TG8h. $\vdash$ SentFuncOne a,b . Vbl c :$\supset$: SentFuncOne (c *qu* a),b.
TG8i. $\vdash$ SentFuncOne a,b . c SF^d_b a :$\supset$: SentFuncOne c,d.
TG8j. $\vdash$ SentFuncOne a,b . b FV a . d SF^c_b a :$\equiv$: SentFuncOne d,c . d FV c . a SF^b_c d.
TG8k. $\vdash$ SentFuncOne a,b . $\sim b$ FV a :$\supset$: Sent a . Vbl b.
TG8l. $\vdash$ c SF^d_b a . $d = b$:$\supset$: $c = a$.

TG9a. $\vdash$ SentFuncOne b',d . SentFuncOne c',e . d' SF^d_e c' :$\supset$: SentFuncOne d',d . SentFuncOne $(b'$ *vee* $d'),d$.

TG9b. $\vdash$ c SF^b_d e . $\sim b$ FV e :$\equiv$: e SF^d_b c . $\sim d$ FV c :$\equiv$: c SF^b_d e . e SF^d_b c.

TG9c. $\vdash$ Sent a . Vbl d . $b = (d$ *qu* $a)$:$\supset$: $\sim d$ FV a . Sent b.

TG9d. $\vdash$ Thm $(e$ *hrsh* $c)$. Fmla e . Fmla c . Thm $(e$ *hrsh tilde* $c)$:$\supset$: (a)(Fmla a .$\supset$. Thm $(e$ *hrsh* $a)$).

TG9e. $\vdash$ Thm $(a$ *hrsh* $b)$. Fmla a . Fmla b . Thm $($*tilde* a *hrsh* $b)$:$\supset$: Thm b.

TG9f. $\vdash$ $(a)(d)$(PredCon $(a \frown$ *invep* $\frown d)$. Vbl a :$\supset$: Thm $(d$ *hrsh* $b))$. Vbl c :$\supset$: $(a)(d)$(PredCon $(a \frown$ *invep* $\frown d)$. Vbl a :$\supset$: Thm $(d$ *hrsh* c *qu* $b))$.

TG9g. $\vdash$ PredCon $($*ex* $\frown$ *invep* $\frown$ *tilde* $\frown$ *lp* $\frown$ *pee* $\frown$ *ex* $\frown$ *vee* $\frown$ *tilde* $\frown$ *pee* $\frown$ *ex* $\frown$ *rp*$)$.

TG9h. $\vdash$Sent a:$\supset$:AtUnivSent a.v.($\sim$AtUnivSent a . UnivSent a).

TG10a. $\vdash$ Sent a . Sent $(a$ *hrsh* $b)$:$\supset$: Sent b.

TG10b. $\vdash$ PredCon a . $\sim$PrimPredCon a :$\equiv$: (Eb)(Ec)(a = ($b \frown$ *invep* $\frown c$) . SentFuncOne c,b).

TG10c. $\vdash$ $a = (b \frown$ *invep* $\frown c)$. SentFuncOne c,b . $a = (d \frown$ *invep* $\frown e)$. SentFuncOne e,d :$\supset$: $b = d$. $c = e$.

TG11. $\vdash$ Fmla a . b Clsr a :$\supset$: Sent b.

TG12a. $\vdash$ Thm a . b Abst a :$\supset$: Thm b.

TG12b. $\vdash$ Thm $($*tilde* $a)$. b Abst a :$\supset$: Thm $($*tilde* $b)$.

TG12c. $\vdash$ Thm a .$\supset$. (Eb)(b Clsr′ a . Fb) :$\supset$: (Sent a . Thm a :$\supset$: Fa).

Every expression a of L is a framed ingredient of some b every framed ingredient of which is a primitive sign of L or is the concatenate of prior framed ingredients of b.

TG13. $\vdash$ (a)(Eb)(a FrIng b . (c)(c FrIng b :$\supset$: PS c .v. (Ed)(Ee) (d Pr_b c . e Pr_b c . $c = (d \frown e)$))).

This rule, the *Rule of Expressions*, will be useful later. The proof is by induction according to *TC3a* and various lemmas concerning 'FrIng' and 'Pr'. It may also be proved by *SynR4*.

Using *TG13* and *TG4* we can prove an Induction Rule slightly stronger than *TC3a*.

TG14. $\vdash$ (a)(PS a .$\supset$. Fa) . $(a)(b)(Fa$. Fb :$\supset$: $F(a \frown b))$:$\supset$: $(a)Fa$.

Finally, an alternative set of axioms for M may be given as follows. We can drop the Rule of Infinite Induction, *SynR4*, and take in its place the Rule of Expressions (*TG13*) and the Rule of Framed Ingredients (*TG4*). On the basis of these together with *SynR1–SynR3* we notice that all of the theorems of this chapter may be proved. We shall retain *SynR4* for most *semantical* purposes in the next few

chapters, but later (in (V,H) and (VIII,D)) we shall dispense with it altogether.

H. Some Further Notions of Syntax. Before turning to semantics proper, let us consider a few further syntactical notions which may be useful later.

The notion of *theorem* has been defined rigorously above (in § F), but not the notion of *proof*. But clearly

'b Proof a' can abbreviate '(a FrIng b . (c)(c FrIng b :⊃: Ax c .v. (Ed)(Ee)(d Pr_b c . e Pr_b c . c IC d,e)) . ∼(Ec)a Pr_b c)'.

An expression b is a proof of a if and only if a is the last or right-most framed ingredient of b, every framed ingredient of which is an axiom or is an immediate consequence of prior framed ingredients of b.

We have immediately that

TH1. ⊢ Thm a .≡. (Eb)b Proof a.

It is often useful to speak not just of proofs, but also of *derivations* from given premisses.[1] That a is derivable from b is defined as follows.

'a Drv b' abbreviates '(Fmla b . (Ec)(a FrIng c . (d)(d FrIng c :⊃: $d = b$ .v. Ax d .v. (Ea')(Eb')(a' Pr_c d . b' Pr_c d . d IC a',b')) . ∼(Ed) a Pr_c d))'.

Note that in a proof we start only with axioms. In a derivation from the premiss b, we start either with axioms or with the premiss b.

If a is a theorem then a is derivable from every formula, and conversely.

TH2. ⊢ Thm a .≡. (b)(Fmla b .⊃. a Drv b).

The proof is immediate.

Occasionally we may wish to speak not only of a derivation from a given premiss b, but of a derivation from a (virtual) class of premisses.

'a Drv $\boldsymbol{F}$' abbreviates '((b) ($\boldsymbol{F}b$.⊃. Fmla b) . (Ec)(a FrIng c . (d) (d FrIng c :⊃: $\boldsymbol{F}d$ .v. Ax d .v. (Ea')(Eb')(a' Pr_c d . $b'$$\mathrm{Pr}_c$ d . d IC a',b')) . ∼(Ed)a Pr_c d))'.

Let 'Λ' abbreviate for the moment 'aɜ∼$a = a$'. Λ is the null (virtual) class of expressions of L. The theorems of L are just those expressions derivable from Λ.

TH3. ⊢ Thm b .≡. b Drv Λ.

[1] The importance of the notion of a derivation has been emphasized especially by Carnap.

Often it is useful to speak of a (virtual) class of formulae as *consistent.*

'Cnst $\boldsymbol{F}$' abbreviates '$(a)(\boldsymbol{F}a :\supset:$ Fmla a . $\sim$(Thm a . Thm (*tilde* a)))'.

A virtual class of formulae is consistent if and only if it contains no formula a such that both a and its negative are provable. (This is a very obvious definition of consistency and seems to be in essential agreement with traditional conceptions. There are other kinds of consistency, some of them syntactical and some semantical. The notion here is purely syntactical and seems appropriate because L contains '$\sim$' and hence where a is a formula of L (*tilde* a) is also.)

One can define consistency of a class of *sentences* by replacing 'Fmla' here by 'Sent'.

We can define the *completeness* of a (virtual) class of *sentences* as follows.

'Cmplt $\boldsymbol{F}$' abbreviates '$(a)(\boldsymbol{F}a :\supset:$ Sent a . (Thm a .v. Thm (*tilde* a)))'.

A virtual class of sentences is complete if and only if given any one of its members a, a is a theorem of L or its negative is.

These exact syntactical notions correspond with the notions of consistency and completeness discussed informally in (I,C).

Let for the moment 'V' abbreviate '$a\ni$Sent a'. V is then the universal (virtual) class of sentences of L. Within M, to say that L is consistent amounts simply to saying that Cnst V or Cnst $a\ni$Fmla a. Similarly 'Cmplt V' says within M that L is complete in the sense defined. For suitable L, these formulae may even be provable within M. But whether provable or not, there are systems L for which both formulae hold. (This circumstance will be useful below, in (IX,G).) There are thus L's which are both consistent and complete, also L's which are consistent but not complete. But there are no L's which are both inconsistent and incomplete.

TH4. $\vdash (a)(\boldsymbol{F}a .\supset.$ Sent $a) . \sim$Cnst $\boldsymbol{F} :\supset:$ Cmplt $\boldsymbol{F}$.

The proof is immediate.

Two formulae are *deductively equivalent* if and only if one is a theorem if and only if the other is also.

'a DedEquiv b' abbreviates '(Fmla a . Fmla b . (Thm a .$\equiv$. Thm b))'.

The following theorem is immediate.

TH5. $\vdash$ Fmla a . Fmla b . Thm (a *tripbar* b) $:\supset:$ a DedEquiv b.

But a and b may be deductively equivalent where (a *tripbar* b) is not

a theorem, so that what is essentially the converse of the theorem stipulated by *TH5* does not hold.

The syntactical notions and theorems which will be useful in the sequel have by no means been exhausted. Rather than to cite them here, we shall give such definitions and theorems below in the places where they are specifically needed.

CHAPTER IV

MULTIPLE DENOTATION: FIRST-ORDER SEMANTICS

In semantics we are interested primarily in certain relations obtaining between the linguistic expressions of a language and the objects or entities which those expressions enable us to talk about. These are the so-called *semantical relations*. Semantics is also concerned fundamentally with the concept of *truth*. Genetically concern seems to have been primarily with the semantical relations, and these have played an important role in the history of the subject. But a rigorous concept of truth is definable in terms of such relations, and thus semantics has come to be the theory of the semantical relations, the theory of truth, and the theory of the interrelation of the two.

For purposes of a deductive development of a formalized semantical meta-language, it seems both natural and convenient to take certain semantical relations as primitive. Several alternative primitives are to be considered in the course of this book. The task of this chapter and the next is the formulation of a first-order semantics for a given L taking a relation of *multiple denotation* as sole semantical primitive. In § A the character of this primitive relation is discussed informally. In § B the *semantical meta-language* of L based upon multiple denotation is sketched in some detail, and *definitions* of various important semantical notions are given in § C. The *axioms* of this semantical meta-language are listed in § D, and some *elementary consequences* thereof in § E. Some further consequences are given in § F, in particular those providing for a kind of *Boolean algebra* whose elements are the predicate constants of L.

A. Multiple Denotation. There are several ways in which one can use 'denotation'. One can speak of an individual, class, or relational constant as denoting its object. One can speak of what one would ordinarily call a class or property word as denoting the members of the class in question. And one can even speak of denotation in a broader sense according to which, e.g., sentences denote propositions or truth-values. We shall use 'denotation' here only in the second of these three senses, according to which predicates are said to denote the objects to which they apply. E.g., 'dog' can be said to denote

severally the particular dogs Marni, Randy, Fido, etc. What are ordinarily regarded as class names thus come to denote severally the members of the class but not the class itself. Denotation construed in this sense is a very simple as well as a very powerful semantical relation. The object-language L here contains no individual constants, and hence there is no need of a denotation relation in the first of the three senses enumerated. (See, however, (X,B).) And because the point of view adopted throughout is wholly extensional, there will be no need of a denotation relation in the third sense.

The view underlying the semantical meta-languages in this chapter and the next is that multiple denotation is the fundamental relation of semantics. Of course this is somewhat arbitrary and for some purposes one would wish to regard other relations as fundamental. But names or expressions which denote (in some sense) seem throughout history to have been the basic elements in semantics, just as the integers have always been in a certain sense the basic elements in mathematics. In terms of multiple denotation, all other semantical notions that are required are definable. The view that multiple denotation is fundamental need not be explicitly argued. An implicit argument will arise from the presentation of the theory, noting its simplicity and generality when compared with some allied formulations of semantics.

Let us use 'Den' to express the primitive relation of multiple denotation. 'Den' is to be the sole semantical primitive of the semantical meta-language for a given object-language L. Now what is it that is to be taken as significantly denoting what? Clearly *expressions* are to be thought of as denoting *objects*. The expressions are to be just those of L and the objects just those of the fundamental domain of L. (The expressions and objects of L may in fact be taken as constituting two mutually exclusive domains, although this is not essential.) 'Den' is therefore to stand for a relation whose domain consists of (certain of) the expressions of L and whose converse domain consists of the objects of L.

Sometimes one uses 'applies' in the sense of multiple denotation. Thus 'man' can be said to apply severally to individual men. Sometimes the phrase 'is true of' is used, or 'is predicable of'. The term 'applies' is already used in logic and mathematics when we speak of the application of a function to an argument, so that it is perhaps better not to use it here in this semantical sense as a technical term. The only objection to 'is true of' is perhaps that it contains 'true' and hence tends to conceal the difference between the semantical relation concerned and the semantical property or concept of truth.

B. Primitive Framework. Let the semantical meta-language of L to be formulated on the basis of 'Den' be called SM_D^L, the semantical meta-language of L based upon multiple denotation.[1]

The structure of SM_D^L must be considerably more complicated than that of the syntactical meta-language M above. Semantics presupposes syntax and contains it as a part, and a semantical meta-language must therefore contain a syntactical one as a part. The syntax language presupposed is M, and hence SM_D^L can in a certain sense be regarded as an extension of M. Within M we can symbolize only relations between expressions of L. Within the semantical meta-language SM_D^L, however, we must also have symbols standing in one way or another for the objects of L. This is essential if we are to be able to express within SM_D^L semantical relations of the kind desired. One way of achieving this is to require that, for every primitive expression of L, there be a corresponding primitive expression within SM_D^L. The symbol of SM_D^L with which a given symbol of L corresponds in this way is called the *translation* of that symbol.[2] And it is to be understood that the concatenate of any two translational expressions of SM_D^L is the translation of the concatenates of the corresponding expressions of L. The correspondence between the expressions of L and their translations within SM_D^L is one-to-one or bi-unique, so that for each expression of L there is one and only one corresponding translation within SM_D^L.

The simplest way of providing for the translational expressions of SM_D^L is no doubt to let each primitive symbol of L be its own translation within SM_D^L. Every sentence or term which can be formed in L can then also be formed in SM_D^L and in precisely the same way. Thus SM_D^L contains all the modes of expression of L. But because SM_D^L contains all the modes of expression of M also, we can, within SM_D^L, talk about or *mention* the expressions of L as well as *use* their translations. To mention the expressions of a language is to talk about the expressions themselves. On the other hand, we use the expressions of a language when we speak about the objects with which that language is concerned. (This distinction we have already encountered above.) In fact, we can mention the expressions of L within SM_D^L in the same way that we do within M. But because of the presence of the additional primitive 'Den' within SM_D^L, we can interrelate the expressions of L with the objects of L by requiring that certain expressions denote certain objects. SM_D^L thus contains

[1] This semantical meta-language is akin to those studied by Tarski. See especially § 5 of *Der Wahrheitsbegriff*, where, however, a predicate for truth rather than 'Den' is taken as primitive.

[2] The word 'translation' is used here in the strict sense of Tarski.

all the modes of expression of M as well as translations of the symbols of L, together with 'Den'.

We see then that SM_D^L contains, roughly speaking, four parts, a logical part, a syntactical part, a translational part, and a semantical part. The logical and syntactical parts are essentially as in M, although the basic logic must be slightly broader. (See below.) But also SM_D^L contains a translation *in toto* of the object-language L. The translational part may be just the object-language L itself or any language which corresponds with L in the way described. Finally, within the semantical part of SM_D^L, we interrelate the expressions of L with their denotata in a suitable way.

Recall that L can be taken as any first-order language with or without identity in either the narrower or wider sense, as described in Chapter II. SM_D^L likewise is to be of first order, but because it contains two distinct styles of variables, it will be of first order only in the sense discussed at the end of (II,H). The variables of each style range over well-specified totalities of entities. The translations of variables of L are to be regarded as ranging over precisely the objects which the variables of L range over. The expressional variables of SM_D^L are simply those of M and range over the expressions of L. Whether SM_D^L be of first order in the narrower or wider sense depends upon L. If L is of first order in the narrow sense, SM_D^L is then also a first-order language in the narrow sense. But if L is of first order in the wider sense, SM_D^L must also be construed in the wider sense. The logical character of SM_D^L depends only upon L, and not upon the syntactical or semantical tools being used. This is an important feature of SM_D^L, in accord with the constructivist tendency. (See (VII,F) and XIII.)

The full specification of the primitive vocabulary of SM_D^L may be summarized as follows.

Primitive Logical Constants: 'v', '$\sim$', '(', ')', '$\mathfrak{z}$', and '='.

Translational Primitives (*Predicate Constants*): 'P', 'P′′', . . . , 'P′′′′′'.

(Variables): 'x', 'x''', 'x'''', . . .

Structural-Descriptive Primitives (*Individual Constants*): *'vee'*, *'tilde'*, *'lp'*, *'rp'*, *'id'*, *'invep'*, *'pee'*, *'ac'*, *'ex'*.

Syntactical Primitive (*Functor*): '$\frown$'.

Expressional Variables: 'a', 'b', 'c', 'd', 'e', 'a''', . . .

Semantical Primitive (*Relational Constant*): 'Den'.

Note that among the translational primitives, we do not include translations of the logical constants of L. These are already provided for by the logical constants of SM_D^L.

For explicitness, let the notion of being a *syntactical term* of SM_D^L be defined informally by recursion within a meta-language of SM_D^L as follows.

1. An expressional variable is a syntactical term of SM_D^L.
2. Any structural-descriptive constant is a syntactical term of SM_D^L.
3. If $\boldsymbol{x}$ and $\boldsymbol{y}$ are syntactical terms, so is '$(\boldsymbol{x} \frown \boldsymbol{y})$'.

Presupposing the notion of a syntactical term, that of *formula* may be defined similarly.

1. Any translational predicate constant of degree n followed by n translational variables, is a formula of SM_D^L. ($1 \leqq n \leqq 5$.)
2. If $\boldsymbol{x}$ and $\boldsymbol{y}$ are syntactical terms, then '$\boldsymbol{x} = \boldsymbol{y}$' is a formula.
3. If $\boldsymbol{x}$ and $\boldsymbol{y}$ are translational variables, then '$\boldsymbol{x} = \boldsymbol{y}$' is a formula.
4. If $\boldsymbol{A}$ is a formula and $\boldsymbol{x}$ and $\boldsymbol{y}$ are translational variables, then '$\boldsymbol{y} \ni \boldsymbol{A}\ \boldsymbol{x}$' is a formula.
5. If $\boldsymbol{A}$ is a formula, $\boldsymbol{x}$ is a syntactical term, and $\boldsymbol{y}$ is an expressional variable, then '$\boldsymbol{y} \ni \boldsymbol{A}\ \boldsymbol{x}$' is a formula.
6. If $\boldsymbol{x}$ is a syntactical term and $\boldsymbol{y}$ is a translational variable, then '$\boldsymbol{x}$ Den $\boldsymbol{y}$' is a formula.
7. If $\boldsymbol{A}$ and $\boldsymbol{B}$ are formulae, so are '$\sim\boldsymbol{A}$' and '$(\boldsymbol{A} \vee \boldsymbol{B})$'.
8. If $\boldsymbol{A}$ is a formula, and $\boldsymbol{x}$ is either an expressional or a translational variable, then '$(\boldsymbol{x})\boldsymbol{A}$' is a formula.

(These two definitions are to be thought of as exhaustive in the sense that nothing is to be regarded as a term or formula of SM_D^L unless explicitly stipulated to be such by them. This statement itself is not simply to be added as another clause to the definitions. The effect of it should rather be provable within a suitable meta-language of SM_D^L.)

Because we shall constantly be writing symbolic expressions *within* SM_D^L and not only *about* SM_D^L, it will be convenient to allow a wider supply of translational variables than we are strictly entitled to. The only translational variables we have are 'x' and '$\boldsymbol{x}$' followed by one or more accents. Let us also use the expressions 'y', 'z', 'w', and 'u', possibly with accents, as additional translational variables. This will make for ease of writing, where we should otherwise have long strings of accents to contend with.

It has been remarked that SM_D^L may be construed as having two mutually exclusive fundamental domains of objects, those of the expressions of L and of the objects of L. This requirement is thought to contribute to clarity and explicitness in discussing SM_D^L and also seems to have a certain naturalness about it. The fundamental objects of most object-languages L, of the kinds met with in the sciences, are clearly of a very different kind from the linguistic

expressions of L. If we construe the two domains of SM_D^L as mutually exclusive, the use of two styles of variables seems appropriate. Note that expressions such as 'a Den b', 'x Den x'', and 'x Den a' are meaningless, as well as expressions such as 'Pa', '$(x \frown rp)$', 'P$'ax$', etc. To construe the objects of L as the expressions of L would result in making use of meaningless expressions. However, this requirement that the two domains be mutually exclusive is by no means essential and, for a general semantical method, it might be thought somewhat artificial. So far as the sequel is concerned, nothing will be gained or lost either way, and one may construe SM_D^L in accord with this requirement or not as he sees fit. Those who are interested primarily in formalized languages for specific areas of science will perhaps favor the requirement. If one is interested primarily in a very broad, comprehensive object-language, in which the whole of science can be expressed or in which a total epistemological or metaphysical system can be formulated, one will no doubt find the requirement too restrictive.

C. Some Definitions. Because the primitives of SM_D^L comprise in part those of M, we can presuppose as definitions of SM_D^L the definitions of M given in Chapter III. The syntactical basis of SM_D^L is thus provided by the elementary theory of concatenation developed there. But because 'Den' is available, several ideas of a semantical kind may now be defined. Many of these ideas have a good deal of intrinsic interest attaching to them. Some will be especially useful in the sequel, and some appear to give definitions of age-old notions in a precise way.

Let us say that an expression a of L *comprehends* an expression b of L if and only if a and b are both predicate constants and a denotes every object which b does.

'a Cmprh b' abbreviates '(PredCon a . PredCon b . $(x)(b$ Den x $.\supset.$ a Den $x)$)'.[1]

The converse of the relation of comprehension will be called semantical *subsumption*.

'a Sub b' abbreviates 'b Cmprh a'.

And, clearly, an expression a which is both subsumed in and comprehends an expression b, is semantically *co-extensive* with it.

'a CoExt b' abbreviates '(a Cmprh b . a Sub b)'.

[1] The term 'comprehends' in what appears to be essentially this sense was used by Hobbes in 1653. See *Leviathan*, p. 15. Cf. the author's "On the Semantics of Hobbes", *Philosophy and Phenomenological Research*, 14 (1954), pp. 205–11.

Suppose for the moment that L contains 'animal' and 'man' as predicate constants. Then 'animal' may be said to comprehend 'man', or 'man' to be subsumed in 'animal'. And if L also contains 'rational', then 'man' and 'rational animal' are presumably co-extensive. The use of 'co-extensive' for this relation of mutual comprehension seems to accord more or less with the traditional usage of this term.

The only kinds of expressions of L we shall regard as denoting expressions are the one-place primitive predicate constant[s] and the one-place abstracts containing no free variables. Clearly, meaningless combinations of symbols of L, such as '$((xx \vee \sim$' should not be regarded as denoting. Nor do the logical constants or sentences denote. To regard them as so doing would require a considerable enrichment in the modes of expression of SM_D^L. Such an enrichment is not needed for present purposes. And certainly such expressions as variables do not denote at all. They *range* over a domain of objects, but cannot properly be said to denote them. (See (VI,F).) Relational predicate constants can be regarded as denoting, but not in the simple sense of which we are here speaking. (Cf. (X,C).) And L contains primitively no individual constants. All we have left, then, are the one-place predicate constants including abstracts. These we shall regard as the denoting expressions of L. Strictly there is no need to introduce a symbol for this notion. To say that a is a denoting expression is simply to say that a is a one-place predicate constant in the sense defined in (III,F).

A predicate constant of L is said to be *common* if and only if it denotes at least two distinct objects.

'Com a' abbreviates '(PredCon a . $(\mathrm{E}x)(\mathrm{E}y)(\sim x = y$. a Den x . a Den $y))$'.

Within L, e.g., '$x \ni (\mathrm{P}x \,.\, (\mathrm{E}x')\mathrm{P}'xx')$' is a common predicate constant, provided 'P' applies to more than one object and the domain of the relation P′ has more than one member. And '$x \ni (\mathrm{P}x \vee \sim \mathrm{P}x)$' is a common predicate constant if the fundamental domain of L contains at least two distinct entities.

A *unit* predicate constant denotes just one object.[1]

'Unit a' abbreviates '(PredCon a . $(\mathrm{E}x)(a$ Den x . $(y)(a$ Den $y \,.\supset.\, x = y)))$'.

If '$(\mathrm{E}x)(\mathrm{P}x \,.\, (x')(\mathrm{P}x' \,.\supset.\, x = x'))$' holds in L, then '$x \ni \mathrm{P}x$' and 'P' are unit predicate constants.

[1] The distinction between common and unit predicate constants is somewhat similar to Woodger's distinction between "shared names" and "unshared names". See his *Biology and Language*, p. 17. See also his "Science without Properties", *The British Journal for the Philosophy of Science*, II, 7 (1951), pp. 193–216. Cf. Hobbes, *loc. cit.*; also Ockham, *Summa Logicae*, I, cc. 68–9.

We must be careful not to confuse unit predicate constants with proper names. A proper name is like a unit predicate constant in denoting uniquely an individual. Unit predicate constants are of course predicate constants, whereas proper names are individual constants. L contains no individual constants primitively and hence primitively no proper names. (For the theory of proper names, see Chapter X below.) Proper names can be introduced within L by definition in terms of descriptions, as we have noted above; but it is of no interest to do so for present purposes.

A *null* predicate constant of L is one which denotes no object.

'Null a' abbreviates '(PredCon a . $\sim$(Ex)a Den x)'.

E.g., '$x\ni$(Px . $\sim$Px)' is a *null* predicate constant of L. And, as one would expect, *universal* predicate constants of L are those which denote every object.

'Univ a' abbreviates '(PredCon a . (x)a Den x)'.

'$x\ni$(Px v $\sim$Px)' is, e.g., a universal predicate constant of L.

Another important semantical notion can be defined as follows. Let '----x----' be any sentential function of the one variable 'x' within the translational part of SM^L_D. Suppose that (i) for all x, an expression a denotes x if and only if ----x----, and (ii) a is a predicate constant. (i) and (ii) give a necessary and sufficient condition under which we may say that *a extends over* or *has as its extension* the totality of objects x such that ----x----. Suppose we write '----x----' as '$x\ni$---x--- x', using the abstraction operator. We could then let

'a Ext $x\ni$---x---' abbreviate '(PredCon a . (x)(a Den x .$\equiv$. $x\ni$---x--- x))'.

Or, using another notation, we let

'a Ext $\boldsymbol{F}$' abbreviate '(PredCon a . (x)(a Den x .$\equiv$. $\boldsymbol{F}x$))',

where $\boldsymbol{F}$ is a primitive one-place predicate constant or a one-place abstract (containing no free variables) within the translational part of SM^L_D. Note that expressions such as $\boldsymbol{F}$ of SM^L_D do not stand in any way for objects which are values for variables. Such expressions are merely abstracts or primitive predicate constants and afford linguistically convenient ways of talking about totalities of objects. These totalities are not classes or properties strictly, not being values for class or property variables. They are what we called above *virtual classes*. They behave very much as though they were classes and have many of the characteristics which classes have. But one cannot quantify over them, one cannot say 'For all virtual classes, such and such holds', and hence they cannot be regarded as genuine classes. The definition just given introduces in effect a kind of virtual-class

denotation. It is not strictly denotation but is closely akin to it. Because a is required by the definition to be a predicate constant, virtual classes may be thought of as whatever it is that one-place abstracts or primitive predicate constants extend over in the sense defined. But strictly there are no such things as virtual classes, and talking about them is therefore tenuous indeed!

Another important semantical relation is that of *satisfaction*, in the restricted sense in which one says that an object x *satisfies* a sentential function a of (at most) one variable. This is definable as follows.

'x Sat a' abbreviates '(Eb)(SentFuncOne a,b . ($b \frown invep \frown a$) Den x)'.

An object x satisfies a if and only if a is a sentential function of some variable b where the abstract ($b \frown invep \frown a$) denotes x.[1] If 'Sat' in this sense were the semantical primitive in place of 'Den', 'Den' would be definable as follows.

'a Den x' abbreviates '(Eb)(Ec)(SentFuncOne b,c . a = ($c \frown invep \frown b$) . x Sat b)'.

A semantics based on satisfaction will be sketched in Chapter VII. In SM_D^L the role of satisfaction is wholly usurped by that of multiple denotation.

Finally, we introduce three important semantical relations akin to the various notions of *logical sum*, *product*, and *negation*.

Suppose for the moment that L contains as predicate constants 'male' and 'female'. We wish then to be able to speak of the complex predicate constant 'male or female'. Let such a complex predicate constant be called a *semantical sum* of the two predicate constants involved. (Analogy with the logical sum of truth-functional logic will be sufficient to justify the terminology.) More precisely, an expression a can be said to be a semantical sum of expressions b and c provided a, b, and c are predicate constants and for all x, a denotes x if and only if b denotes x or c denotes x.

'a Sum b,c' abbreviates '(PredCon a . PredCon b . PredCon c . (x) (a Den x :≡: b Den x .v. c Den x))'.

Within L, to give another example, '$x \ni$(Px v (Ex')P$'xx'$)' is a semantical sum of 'P' and '$x \ni$(Ex')P$'xx'$', because all three of these expressions are predicate constants and the first denotes just the objects denoted by the second or the third.

In a similar way one can introduce the notion of a *semantical product*. The product of two predicate constants is to denote just those objects denoted by both of the predicate constants.

[1] This satisfaction relation is very much simpler than those studied by Tarski, *Der Wahrheitsbegriff*, p. 311 and throughout.

'a Prod b,c' abbreviates '(PredCon a . PredCon b . PredCon c . (x) (a Den x :≡: b Den x . c Den x))'.

Within L, to give an example, '$x\ni(\mathrm{P}x\ .\ (\mathrm{E}x')\mathrm{P}'xx')$' is a semantical product of '$x\ni\mathrm{P}x$' and '$x\ni(\mathrm{E}x')\mathrm{P}'xx'$'.

Finally, given any predicate constant we wish to be able to form its *semantical negative*. The one is to denote precisely the objects not denoted by the other.

'a Neg b' abbreviates '(PredCon a . PredCon b . (x)(a Den x .≡. ∼b Den x))'.

Within L, e.g., '$x\ni\mathrm{P}x$' and '$x\ni\sim\mathrm{P}x$' are clearly semantical negatives of each other. A predicate constant and a semantical negative of it have mutually exclusive denotata, but are jointly exhaustive in the sense that their semantical sum denotes everything.

Notions akin to those of semantical sum, product, and negative have occasionally been discussed in writings on traditional semantics. The notions have always been somewhat vague, however, and scarcely any attempt has been made to provide for them a precise semantical framework.

D. Rules of Semantics. Let us now consider the axiomatic structure of SM^L_D. There will be four kinds of axioms, corresponding to the four parts of the semantical meta-language distinguished above. These will include (1) logical axioms, (2) non-logical or translational axioms, (3) syntactical axioms, and finally (4) some semantical axioms.

The *Logical Axioms* concerning truth-functions are given by *R1–R4*, of (II,A), where the letters '$\boldsymbol{A}$', etc., of those rules stand for any formulae of SM^L_D. The Quantification Rule *R5* must, however, be expressed as follows.

R5'. $\vdash (\boldsymbol{x})\boldsymbol{A} \supset \boldsymbol{B}$, where $\boldsymbol{x}$ is an expressional variable and $\boldsymbol{y}$ is a syntactical term, or $\boldsymbol{x}$ and $\boldsymbol{y}$ are both translational variables, and (etc., as in *R5* about $\boldsymbol{A}$ and $\boldsymbol{B}$).

In *R6*, '$\boldsymbol{A}$' and '$\boldsymbol{B}$' are to stand for any formulae and '$\boldsymbol{x}$' for any translational or expressional variable of SM^L_D. Likewise *MP* and *Gen* becomes rules of SM^L_D if we construe '$\boldsymbol{A}$' and '$\boldsymbol{B}$' as standing for any formulae of SM^L_D and '$\boldsymbol{x}$' for any expressional or translational variable. The rule providing for the *Axioms of Abstraction* must have the proviso that either (i) $\boldsymbol{x}$ and $\boldsymbol{y}$ are both translational variables or (ii) $\boldsymbol{x}$ is a syntactical term and $\boldsymbol{y}$ is an expressional variable, $\boldsymbol{y}$ is not free in $\boldsymbol{A}$, and (etc., about $\boldsymbol{A}$ and $\boldsymbol{B}$ as required). The identity rules *R7* and *R8* also become rules of SM^L_D, if the exhibited variables

involved are both taken as either syntactical terms or as translational variables of SM_D^L. These logical rules of SM_D^L will be referred to subsequently as *R1′–R8′*, *MP′*, *Gen′*, and *Abst′*.

The *Translational Axioms* of SM_D^L comprise just the translations of the non-logical axioms of L. These will ordinarily be axioms appropriate to the particular discipline formalized by L. The *Syntactical Axioms* of SM_D^L comprise just those of M, stipulated by *SynR1–SynR4* (II,B).

Because of the presence of 'Den' within SM_D^L, we must have in addition some *Rules of Denotation* or *Semantical Rules*.[1]

We must assure somehow or other that within SM_D^L abstracts such as '$x\ni$---x---' extend over the corresponding virtual classes. Thus we should wish

(1) '$(x)(a \text{ Den } x .\equiv. \text{---}x\text{---})$'

to hold, where in place of 'a' we put in the structural description of the abstract '$x\ni$---x---', '---x---' being any sentential function of the one variable 'x' within the translational part of SM_D^L. The simplest way of assuring that all expressions such as (1) hold within SM_D^L seems to be to assume them outright by means of a rule. Thus the first *Rule of Denotation*, stipulating infinitely many axioms of SM_D^L, is as follows.

DenR1. $\vdash (\boldsymbol{x})(a \text{ Den } \boldsymbol{x} .\equiv. \text{---}\boldsymbol{x}\text{---})$, where (i) '---$\boldsymbol{x}$---' is any sentential function of the one variable $\boldsymbol{x}$ in the translational part of SM_D^L, and in place of 'a' we put in the structural description of the abstract '$\boldsymbol{x}\ni$---$\boldsymbol{x}$---', or (ii) in place of 'a' we put in '*pee*' and '---$\boldsymbol{x}$---' is the sentential function '$\mathrm{P}\boldsymbol{x}$'.

(This second clause is required because 'P' is the only one-place primitive predicate constant of L.) According to this rule, e.g.,

$\vdash (x)((ex \frown invep \frown lp \frown pee \frown ex \frown vee \frown tilde \frown pee \frown ex \frown rp) \text{ Den } x .\equiv. (\mathrm{P}x \vee \sim\mathrm{P}x))$.

The only additional axiom needed is stipulated by a *Restrictive Rule*. This states that if an expression a of L denotes any x at all, a is a predicate constant.

DenR2. $\vdash a \text{ Den } x . \supset . \text{PredCon } a$.

But of course the converse of the formula stipulated by *DenR2* does not hold, because null predicate constants denote no object. We should know from the definitions of 'Com', 'Unit', 'Univ', and 'Null' above (in § C) that nonsense expressions such as '$\mathrm{vv}\sim x$', etc., are not common, unit, universal, or null predicate constants. But we should

[1] Cf. Tarski, *Der Wahrheitsbegriff*, pp. 305–6, conditions (α) and (β).

not know that they do not denote at all, except by postulating *DenR2* or some equivalent.

To summarize. The list of rules of SM_D^L comprises *R1'–R8'*, *MP'*, *Gen'*, *Abst'*, *translations of the non-logical axioms of L*, *SynR1–SynR4*, and *DenR1–DenR2*.

E. Some Elementary Theorems. Let us now note some elementary consequences of the axioms and rules of SM_D^L. The purely logical theorems have in effect already been proved in Chapter II, and the syntactical part of SM_D^L has been considered in detail in Chapter III. So here we may concern ourselves exclusively with semantical theorems, those involving 'Den' in an essential way or which depend upon *DenR1* or *DenR2*.

The first theorem states that a certain predicate constant denotes all the objects of *L*.

TE1a. $\vdash (x)(ex \frown invep \frown lp \frown pee \frown ex \frown vee \frown tilde \frown pee \frown ex \frown rp)$ Den x.

The proof is by *DenR1*, which states in particular that

$$\vdash (x)((ex \frown invep \frown \ldots \frown rp) \text{ Den } x \mathrel{.\equiv.} (\mathrm{P}x \vee \sim \mathrm{P}x)).$$

Hence also

TE1b. $\vdash (\mathrm{E}a)(x)a$ Den x.

Similarly, one proves that

TE1c. $\vdash \sim(\mathrm{E}x)(ex \frown invep \frown tilde \frown lp \frown pee \frown ex \frown vee \frown tilde \frown pee \frown ex \frown rp)$ Den x.

And hence also that

TE1d. $\vdash (\mathrm{E}a)(x)\sim a$ Den x.

Recalling *TG9g*(III) that (taking the abstract of *TE1c*),

$$\vdash \text{PredCon } (ex \frown invep \frown tilde \frown \ldots \frown rp),$$

we also have that

TE1e. $\vdash (\mathrm{E}a)(\text{PredCon } a \mathrel{.} (x)\sim a \text{ Den } x)$.

The following theorem is frequently useful and is a corollary to *DenR1*.

TE2. $\vdash (y)(a \text{ Den } y \mathrel{.\equiv.} A)$, where '$a$' is the structural description of an abstract '$\boldsymbol{x}\ni$---$\boldsymbol{x}$---' (from the translational part of the metalanguage) containing no free variables, y is a variable distinct from $\boldsymbol{x}$ (and not occurring in '---$\boldsymbol{x}$---'), and A differs from '---$\boldsymbol{x}$---' only in containing y wherever (and only where) there are free occurrences of $\boldsymbol{x}$ in '---$\boldsymbol{x}$---'.

The proof is immediate by *DenR1* and a quantificational law.

The following theorem is a lemma concerning unit predicate constants, more specifically, a *uniqueness* law.

TE3. $\vdash$ Unit a . a Den x . a Den y :$\supset$: $x = y$.

This is proved immediately by noting that

$\vdash$ (Ex)(a Den x . (y)(a Den y . $\supset$. $x = y$)) . a Den z . a Den w :$\supset$: $z = w$.

The next theorem is the denial of a principle of uniqueness on the hypothesis that the fundamental domain of L contains at least two distinct objects.

TE4. $\vdash$ (Ex)(Ey)$\sim x = y$.$\supset$. $\sim$(a)(x)(y)(a Den x . a Den y :$\supset$: $x = y$).

By *TE1a*,

$\vdash$ (Ex)(Ey)$\sim x = y$.$\supset$. (Ex)(Ey)(($ex \frown invep \frown \ldots \frown rp$) Den x . ($ex \frown invep \frown \ldots \frown rp$) Den y . $\sim x = y$).

Then we gain *TE4* using some quantificational laws.

To prove the denial of the uniqueness principle we should wish to drop the hypothesis of *TE4* outright. This hypothesis we cannot prove separately except by appeal to the non-logical axioms of L. Ordinarily the non-logical axioms should enable us to prove this readily. Rather than to assume that they do, however, we prefer to take '(Ex)(Ey)$\sim x = y$' as hypothesis wherever needed. And also, for other theorems to be given, we shall simply add an appropriate hypothesis where needed rather than to assume that it is a consequence of the non-logical axioms of L.

The following theorems are principles of identity.

TE5a. $\vdash x = y$.$\supset$. (a)(a Den x .$\equiv$. a Den y).

This is an immediate consequence of *R8′*. The converse also holds on a suitable hypothesis.

TE5b. $\vdash$ (Ea)(Unit a . a Den x) :$\supset$: (a)(a Den x .$\equiv$. a Den y) .$\supset$. $x = y$.

TE5c. $\vdash a = b$.$\supset$. (x)(a Den x .$\equiv$. b Den x).

This follows also immediately from *R8′*. The converse is not provable.

TE6. $\vdash$ PredCon a . PredCon b :$\supset$: a CoExt b .$\equiv$. (x)(a Den x .$\equiv$. b Den x).

The proof is immediate using the definition of 'CoExt'.

Clearly the relation of comprehension is *transitive* and *reflexive* if confined to predicate constants.

TE7a. $\vdash a$ Cmprh b . b Cmprh c :$\supset$: a Cmprh c.

TE7b. $\vdash$ PredCon a .$\supset$. a Cmprh a.

The proofs are immediate. Hence also,

TE7c. $\vdash$ PredCon a . PredCon b :$\supset$: a Cmprh b .$\equiv$. (c)(PredCon c . b Cmprh c :$\supset$: a Cmprh c).

The proof is by *TE7a* and *TE7b*.

We have already noted one important law concerning co-extensiveness, *TE6*. We have also the following additional co-extensiveness laws.

TE8a. $\vdash$ PredCon a .$\supset$. a CoExt a.

TE8b. $\vdash$ a CoExt b .$\equiv$. b CoExt a.

TE8c. $\vdash$ a CoExt b . b CoExt c :$\supset$: a CoExt c.

TE8d. $\vdash$ PredCon a :$\supset$: a CoExt b .$\equiv$. (c)(c CoExt a .$\equiv$. c CoExt b).

The following is a kind of uniqueness law concerning the extension of expressions for virtual classes. It states that if a and b both extend over the same virtual class, then a and b are co-extensive.

TE9a. $\vdash$ a Ext $\boldsymbol{F}$. b Ext $\boldsymbol{F}$:$\supset$: a CoExt b.

The proof is immediate from the definition of 'Ext', using *TE6*. Also we have that

TE9b. $\vdash$ a Ext $\boldsymbol{F}$. a Ext $\boldsymbol{G}$:$\supset$: $(x)(\boldsymbol{F}x$.$\equiv$. $\boldsymbol{G}x)$,

TE9c. $\vdash$ a Ext $\boldsymbol{F}$. a CoExt b :$\supset$: b Ext $\boldsymbol{F}$,

and

TE9d. $\vdash$ PredCon a :$\supset$: a Ext $\boldsymbol{F}$.$\equiv$. (b)(b CoExt a .$\supset$. b Ext $\boldsymbol{F}$).

Note that the only expressions which extend over virtual classes are, by definition, predicate constants. We have not only that

TE10a. $\vdash$ a Ext $\boldsymbol{F}$.$\supset$. PredCon a,

but also, using *DenR2*, that

TE10b. $\vdash$ a Den x .v. a Ext $\boldsymbol{F}$:$\supset$: PredCon a.

We have another co-extensiveness law as follows.

TE11. $\vdash$ PredCon a . Vbl b :$\supset$: a CoExt $(b \frown invep \frown a \frown b)$.

The proof is by *Abst'*, *DenR1*, *SynR4*, and *TE6*.

Finally, we note the following theorems concerning *vee* and *tilde*.

TE12. $\vdash$ SentFuncOne a,c . SentFuncOne b,c :$\supset$: $(x)((c \frown invep \frown (a\ vee\ b))$ Den x .$\equiv$. $((c \frown invep \frown a)$ Den x .v. $(c \frown invep \frown b)$ Den $x))$.

TE13. $\vdash$ SentFuncOne a,b .$\supset$. $(x)((b \frown invep \frown a)$ Den x.$\equiv$.$\sim(b \frown invep \frown tilde\ a)$ Den $x)$.

The proofs are by *DenR1* and *SynR4*.

F. Boolean Algebra of Predicate Constants. Let us now note some further properties of semantical sums, products, and negatives, before introducing the semantical concept of truth in the next chapter. In particular, it will be observed that the theory of predicate

constants forms a *Boolean algebra* within SM^L_D, if we take the Boolean relation of inclusion as semantical subsumption, the Boolean identity as co-extensiveness, the Boolean addition, multiplication, and negation respectively as the notions of forming the semantical sum, product, and negation, and the Boolean universal and null elements respectively as universal and null predicate constants. Thus we have the following theorems in SM^L_D.[1]

TF1. $\vdash$ PredCon a .$\supset$. a Sub a.

TF2. $\vdash$ a Sub b . b Sub c :$\supset$: a Sub c.

TF3. $\vdash$ a CoExt b .$\equiv$. (a Sub b . b Sub a).

TF4. $\vdash$ PredCon b . PredCon c :$\supset$: (Ea)(a Sum b,c . (d)(d Sum b,c .$\supset$. d CoExt a)).

TF5. $\vdash$ a Sum b,c :$\supset$: b Sub a . c Sub a.

TF6. $\vdash$ d Sum a,b .$\supset$. (c)(a Sub c . b Sub c :$\supset$: d Sub c).

TF7. $\vdash$ PredCon b . PredCon c :$\supset$: (Ea)(a Prod b,c . (d)(d Prod b,c .$\supset$. d CoExt a)).

TF8. $\vdash$ a Prod b,c :$\supset$: a Sub b . a Sub c.

TF9. $\vdash$ d Prod a,b .$\supset$. (c)(c Sub a . c Sub b :$\supset$: c Sub d).

TF10. $\vdash$ a' Prod a,b . b' Prod a,c . e' Sum a',b' . d Sum b,c . e Prod a,d :$\supset$: e CoExt e'.

TF11. $\vdash$ a' Sum a,b . b' Sum a,c . e' Prod a',b' . d Prod b,c . e Sum a,d :$\supset$: e CoExt e'.

TF12. $\vdash$ (Ea)(Univ a . (b)(Univ b .$\supset$. b CoExt a)).

TF13. $\vdash$ (Ea)(Null a . (b)(Null b .$\supset$. b CoExt a)).

TF14. $\vdash$ PredCon a .$\supset$. (Eb)(b Neg a . (c)(c Neg a .$\supset$. c CoExt a)).

TF15. $\vdash$ a Neg b . c Prod a,b :$\supset$: Null c.

TF16. $\vdash$ a Neg b . c Sum a,b :$\supset$: Univ c.

In the statement of these laws, it might be thought that additional hypotheses are required to the effect that the expressions (of L) concerned are predicate constants. But these are not needed in view of the following law, which states that the relata of the relations of being a semantical sum, product, or negation are predicate constants.

TF17. $\vdash$ a Sum b,c .v. a Prod b,c .v. (a Neg b . PredCon c) :$\supset$: PredCon a . PredCon b . PredCon c.

Any further Boolean theorems needed we may presuppose as proved from *TF1–TF17*.

[1] Cf. Tarski, "Zur Grundlegung der Boole'schen Algebra I", *Fundamenta Mathematicae*, 24 (1935), pp. 177–98, especially p. 178. Cf., e.g., also the still useful L. Couturat, *The Algebra of Logic* (Open Court Publishing Co., Chicago and London: 1914), and G. Birkhoff and S. MacLane, *A Survey of Modern Algebra* (Macmillan, New York: 1941).

The significance of the Boolean algebra of predicate constants will become evident when we turn to a semantics of a different kind in Chapters VIII–IX. There such a theory will play an important role. Some of these theorems, as well as some of their further consequences, will also be useful in Chapters V–VII.

CHAPTER V

TRUTH, ADEQUACY, AND CONSISTENCY

THIS chapter is concerned primarily with the concept of truth, one of the most important concepts of semantics. In § A and § B several *alternative definitions* of being true in a given object-language L will be given within SM_D^L. A proof that the truth-definitions are *adequate* in a certain sense will be given in two ways in § C. In § D some informal comments concerning the significance of the adequacy proofs will be offered. The important connection between the semantical truth-concept and the notion of *consistency* is indicated in § E. In § F some *further notions* of semantics are introduced. In § G a proof of the *relative consistency* of SM_D^L is given. It is there shown that the semantical axioms of SM_D^L are consistent on a suitable assumption. In § H, different types of *self-reference* are discussed and it is shown that SM_D^L cannot be interpreted in such a way as to contain its own semantics. Finally, in § I, the important connection between the truth-concept and the notion of *undecidability* is indicated roughly. The material in §§ H–I is a little more difficult mathematically than that which precedes it and may be omitted if desired. There are only occasional references to this material subsequently.

A. Truth. Recall that an atomic universal sentence of L is one of the form '(x)---x---' where '---x---' is a sentential function of the one variable 'x'. And universal sentences of L are sentences which we form by a finite number of applications of disjunction and negation starting with atomic universal sentences. It is convenient to subdivide the first definition of truth in two parts, one as applied to atomic universal sentences, the other as applied to non-atomic universal sentences.

Under what circumstances could we say, in terms of denotation, that an atomic universal sentence '(x)---x---' is *true*? Some reference to *every* object would seem essential as well as to just those objects x such that ---x---. For the definition we should need an expression c denoting all objects of L and an expression b denoting just those objects x such that ---x---. But how are c and b to be interrelated? Suppose we require that b comprehend c. Because c is universal, b is then also (IV,F). Hence, we can say that '(x)---x---'

is true in L if and only if '$x\ni$---x---' is a universal predicate constant of L.

For the precise statement of this truth-definition, the following abbreviation is useful. Let

'$c\ \mathrm{R}_{rp}^{invep}\ d$' abbreviate '(E$a$)(E$b$)(SentFuncOne $a,b\,.\,c = (b \frown invep \frown a)\,.\,d = (b\ qu\ a)$)'.

'$c\ \mathrm{R}_{rp}^{invep}\ d$' holds just where c is an abstract and d the corresponding atomic universal sentence. The 'R' here is to suggest 'replacement'. The word seems apt because we form c from d by deleting the left-most parenthesis of d and replacing the left-most rp of d by $invep$.

We may then say that an atomic universal sentence a is true in L if and only if there is an abstract b which bears R_{rp}^{invep} to a and which is universal.

(I) '$\mathrm{Tr}_1\ a$' abbreviates '(AtUnivSent a . (Eb)(Univ b . $b\ \mathrm{R}_{rp}^{invep}\ a$))'.

(The subscript '1' here is simply to indicate that this is a first truth-predicate, a second to follow.)

We now define the truth-concept as applied to *non*-atomic universal sentences of L. Recall from (II,G) that any universal sentence can be brought into an equivalent conjunctive normal form. The conjunctive normal forms are conjunctions of disjunctions, each minimal component of each disjunction being an atomic constituent or the negative of one. Under what circumstances would we wish to say that a (non-atomic) universal sentence is true? Rather than to consider the sentence itself, let us consider a conjunctive normal form for it. This will be a conjunction of disjunctions and hence will be true only where all of the conjuncts are true. But each minimal conjunct in turn is a disjunction and hence will be true only where at least one of the minimal disjuncts is true. And each minimal disjunct is either an atomic universal sentence or the negation of one. Therefore, we can say that a non-atomic universal sentence c is true if and only if there exists a sentence a which is a conjunctive normal form for c, and for all minimal conjuncts b of a, there exists an atomic universal sentence d as a minimal disjunct of b and d is true (in the sense of 'Tr_1') or else there exists an e such that e is the negative of an atomic universal sentence d, e being a minimal disjunct of b, and d is not true (in the sense of 'Tr_1'). Roughly, this statement can be taken as a definition. In order to make it precise, we need several more definitions of a syntactical kind.

First we need a formal definition of the notion of being a *conjunctive normal form of*. This notion was discussed only informally in (II,G). We confined ourselves there to universal sentences. We shall

do so here also, these being the only kinds of sentences for which the notion of a conjunctive normal form is needed.

An expression *a* will be called a *disjunction* of atomic universal sentences or their negations if and only if it is built out of atomic universal sentences or their negations wholly by means of 'v'. This we can define by means of framed ingredients.

'Disj *a*' abbreviates '(E*b*)(*a* FrIng *b* . (*c*)(*c* FrIng *b* :⊃: AtUnivSent *c* .v. (E*d*)(AtUnivSent *d* . *c* = (*tilde d*)) .v. (E*d*)(E*e*)(*d* Pr_b *c* . *e* Pr_b *c* . *c* = (*d vee e*))))'.

An expression *a* will be called a *minimal disjunct* of *b* provided *b* is a disjunction of atomic universal sentences or their negations, *a* is an atomic universal sentence or the negation of one, and *a* is a segement of *b*.

'*a* MinDisj *b*' abbreviates '(Disj *b* . (AtUnivSent *a* .v. (E*c*)(AtUnivSent *c* . *a* = (*tilde c*))) . *a* Seg *b*)'.

An expression *a* will be called a *conjunction* (of a certain kind) provided it is a framed ingredient of an expression *b*, every framed ingredient of which is either a disjunction of atomic universal sentences or their negations or is constructed from prior ingredients by means of conjunction.

'Conj *a*' abbreviates '(E*b*)(*a* FrIng *b* . (*c*)(*c* FrIng *b* :⊃: Disj *c* .v. (E*d*)(E*e*)(*d* Pr_b *c* . *e* Pr_b *c* . *c* = (*d dot e*))))'.

An expression *a* will be called a *minimal conjunct* of *b* if and only if *b* is a conjunction (in the sense defined), *a* is a disjunction (of atomic universal sentences or their negations), and *a* is a segment of *b* but is a segment of no disjunction other than itself.

'*a* MinConj *b*' abbreviates '(Conj *b* . Disj *a* . *a* Seg *b* . ∼(E*c*)(Disj *c* . ∼*c* = *a* . *a* Seg *c*))'.

Bearing these ideas in mind, we can define the notion of being a conjunctive normal form of, in the sense in which we require it for the definition of truth for universal sentences. An expression *a* is a (*special*) *conjunctive normal form* of *b*, namely, if (i) *a* is a conjunction in the sense defined, (ii) for every atomic universal sentence *c*, if *c* is a segment of *a* then *c* is a segment of *b* and conversely, and (iii) (*a tripbar b*) is a logical theorem. Hence

'*a* CNF *b*' can abbreviate '(Conj *a* . (*c*)(AtUnivSent *c* :⊃: *c* Seg *a* .≡. *c* Seg *b*) . LogThm (*a tripbar b*))'.

The condition here requires that *a* be constructed out of the atomic universal sentences of *b* (or the negatives of such). (One could also require that each atomic universal sentence of *b* or its negative

reappear as a minimal disjunct of *each* minimal conjunct of a. But this is not essential.)

The precise definition of truth as applied to non-atomic universal sentences of L, given above informally, may now be given as follows.

(II) '$\mathrm{Tr}_2 a$' abbreviates '(UnivSent a . ~AtUnivSent a . (Eb)(b CNF a . (c)(c MinConj b :⊃: (Ed)(AtUnivSent d . d MinDisj c . Tr_1 d) .v. (Ed)(Ee)(AtUnivSent d . e MinDisj c . e = (*tilde* d) . ~Tr_1 d))))'.

The definitions (I) and (II) together constitute a general definition of truth for all sentences of L. Hence we can let

'Tr a' abbreviate '(Tr_1 a .v. Tr_2 a)'.

Each sentence of L is either an atomic universal sentence or a non-atomic universal sentence and hence, if true, is true in either the sense of 'Tr_1' or of 'Tr_2'.

B. Some Alternative Definitions. Let us turn now to some alternatives to the definition above.

The first alternative is to be gained by reflection upon the Prenex Normal Form Theorem. Recall that sentences in prenex normal form are of the form

'$(Qx_1) \ldots (Qx_n)A$',

where $x_1, \ldots, x_n$ are distinct variables occurring in A, A contains no quantifiers and no free variables other than $x_1, \ldots, x_n$, and each '(Qx_i)' is either '(x_i)' or '(Ex_i)'. The prenex normal form theorem states that given any sentence B of L there exists at least one sentence C in prenex normal form such that $\vdash C \equiv B$. Also we can require, without loss of generality, that C begin with a universal quantifier, as we have already noted in (II,G). Hence C is an atomic universal sentence. These observations suggest another truth-definition, using prenex normal forms in place of conjunctive normal forms. A sentence a of L will be true, namely, if and only if there exists a sentence b such that b is a prenex normal form of a and is true in the sense of (I).

To make this definition precise, we may presuppose a definition of the notion of being a prenex normal form of. Let 'a PNF b' express that a is a prenex normal form of b beginning with a universal quantifier. We may then define truth as follows.

(III) 'Tr a' abbreviates '(Sent a . (Eb) (Ec)(b PNF a . Univ c . c R_{rp}^{invep} b))'.

Note that this definition is not just an alternative to (I). It is rather an alternative to (I) and (II) together.

A simpler definition of truth is as follows.

(IV) 'Tr a' abbreviates '(Sent a . (Eb)(Vbl b . Univ ($b \frown invep \frown (a$ *dot* $b \frown id \frown b)))$)'.

A sentence '------' of L is true according to this definition if and only if any abstract of the form '$x\ni$(------. $x = x$)' is universal.

A still simpler definition is to be gained by the use of vacuous abstracts. Any abstract of SM_D^L '$x\ni$------' is vacuous if '------' is a sentence because the variable x cannot occur freely in it. '------' can then be regarded as true if and only if there is at least one corresponding vacuous abstract of the form '$x\ni$------' which is universal.

(V) 'Tr a' abbreviates '(Sent a . (Eb)(Vbl b . Univ ($b \frown invep \frown a$)))'.

This provides an especially simple alternative to the three definitions given.

We shall have a basis for comparing these various definitions when we have considered their *adequacy*.

C. Proof of Adequacy. A definition of truth for L is said to be *adequate*, or the concept therewith defined as adequate, if and only if every formula of the meta-language SM_D^L of the form

(T) 'Tr a .≡. ------'

is provable in SM_D^L where in place of 'a' we put in the structural description of '------', '------' being any sentence from the translational part of SM_D^L.[1]

Our task now is to prove within SM_D^L that the truth-concept for L, in one or more of the senses defined, is adequate. We take first the truth-concept as defined in terms of prenex normal forms. The proof of adequacy on the basis of this definition is especially simple and straightforward. It amounts to proving the *First Adequacy Rule*:

TC1. ⊢ Tr a .≡. ------, where 'a' is taken as the structural description of '------', '------' is a sentence from the translational part of SM_D^L, and 'Tr' is defined by (III) in § B above.

The proof is in two parts, the first showing the conditional from left to right, and the second, from right to left. We first show that

(1) ⊢ Tr a .⊃. ------,

where (etc.). Using the Prenex Normal Form Theorem we know that

⊢ (x)---x--- .≡. ------,

if '(x)---x---' is a prenex normal form of '------'. The following lemma is also useful.

[1] Cf. A. Tarski, *op. cit.*, pp. 305–6, and R. Carnap, *Introduction to Semantics*, pp. 26–9. Note that in requiring the *provability* of formulae of the form (T), this definition of adequacy perhaps differs from those of Tarski and Carnap.

Lemma C1. If 'b' is taken as the structural description of the sentence '(x)---x---', from the translational part of SM^L_D, then

$$\vdash c\ \mathrm{R}^{invep}_{rp}\ b\ .\supset.\ (x)(c\ \mathrm{Den}\ x\ .\equiv.\ \text{---}x\text{---}).$$

(This lemma holds in virtue of the first Rule of Denotation (*DenR1*) and *SynR4*. If 'c' is taken as the structural description of '$x\ni$---x---' then obviously the lemma holds. If 'c' is taken as any other structural description, '$c\ \mathrm{R}^{invep}_{rp}\ b$' fails and hence the lemma holds. Then, using *SynR4*, we know that the lemma holds for variable 'c'.) Using *Lemma C1*, we have that

$$\vdash b\ \mathrm{PNF}\ a\ .\ \mathrm{Univ}\ c\ .\ c\ \mathrm{R}^{invep}_{rp}\ b :\supset: (x)(c\ \mathrm{Den}\ x\ .\equiv.\ \text{---}x\text{---}),$$

if (etc., as in the definition of 'adequacy'), and 'b' is taken as the structural description of '(x)---x---'. Hence also because 'Univ c' is an hypothesis, we have that

$$\vdash b\ \mathrm{PNF}\ a\ .\ \mathrm{Univ}\ c\ .\ c\ \mathrm{R}^{invep}_{rp}\ b :\supset: (x)\text{---}x\text{---},$$

where (etc., about 'a' and 'b'). Hence

(2) $$\vdash b\ \mathrm{PNF}\ a\ .\ \mathrm{Univ}\ c\ .\ c\ \mathrm{R}^{invep}_{rp}\ b :\supset: \text{------},$$

where (etc., about 'a') and 'b' is taken as the structural description of some prenex normal form of '------'. But (2) also holds where 'b' is taken as a structural description of any expression which is not a prenex normal form of '------', because in this case

$$\vdash \sim b\ \mathrm{PNF}\ a.$$

Hence (2) holds for 'b' taken as any structural description whatsoever. Using *SynR4* we then immediately have (1).

For the second part of the proof we show that

(3) $$\vdash \text{------}\ .\supset.\ \mathrm{Tr.}\ a,$$

where 'a' is taken as the structural description of '------'. Let '(x)---x---' be taken as above as a prenex normal form of '------'. Then

$$\vdash \text{------}\ .\ b\ \mathrm{PNF}\ a\ .\ c\ \mathrm{R}^{invep}_{rp}\ b :\supset: b\ \mathrm{PNF}\ a\ .\ c\ \mathrm{R}^{invep}_{rp}\ b\ .\ (x)\text{---}x\text{---},$$

where 'a' is taken as the structural description of the sentence '------'. By *Lemma C1*, we have that

(4) $$\vdash \text{------}\ .\ b\ \mathrm{PNF}\ a\ .\ c\ \mathrm{R}^{invep}_{rp}\ b :\supset: b\ \mathrm{PNF}\ a\ .\ c\ \mathrm{R}^{invep}_{rp}\ b\ .\ (x)\text{---}x\text{---}\ .\ (x)(c\ \mathrm{Den}\ x\ .\equiv.\ \text{---}x\text{---}),$$

where (as above about 'a' and 'b'). From syntax

(5) $$\vdash b\ \mathrm{PNF}\ a\ .\supset.\ \mathrm{AtUnivSent}\ b,$$

and

$$\vdash \mathrm{AtUnivSent}\ b\ .\ c\ \mathrm{R}^{invep}_{rp}\ b :\supset: \mathrm{PredCon}\ c.$$

Thus from (4) we have that

$$\vdash \text{------} \,.\, b \text{ PNF } a \,.\, c \; \mathrm{R}^{invep}_{rp} \; b :\supset: b \text{ PNF } a \,.\, c \; \mathrm{R}^{invep}_{rp} \; b \,.\, \text{Univ } c,$$

where (as in (4)), using the definition of 'Univ'. From this we have that

(6) $$\vdash \text{------} \,.\, b \text{ PNF } a \,.\, c \; \mathrm{R}^{invep}_{rp} \; b :\supset: \text{Tr } a,$$

where (as above), using the definition of 'Tr' in terms of 'PNF'. Now from syntax we can obviously prove that

$$\vdash \text{AtUnivSent } b \,.\supset.\, (\mathrm{E}c)c \; \mathrm{R}^{invep}_{rp} \; b.$$

Using this together with (5), we have from (6) that

$$\vdash \text{------} \,.\, b \text{ PNF } a :\supset: \text{Tr } a,$$

where (as above). But in view of the hypotheses about 'a' and 'b', we know that

$$\vdash b \text{ PNF } a.$$

Using this, we gain (3). (3) and (1) together give a proof of the *First Adequacy Rule*.

The proof of adequacy on the basis of (I) and (II) is somewhat more complicated. It requires several lemmas concerning conjunctive normal forms. This proof is omitted. The statement that the truth-concept as defined by (I) and (II) is adequate is called the *Second Adequacy Rule* (*TC2*).

The proofs of the *Third* and *Fourth Adequacy Rules* (*TC3a* and *TC3b*) for the definitions (IV) and (V) respectively are very easy. That for (V) may be given as follows. Using *DenR1* (or *TE2*(IV)) and *SynR4* we have immediately that

Lemma C2. $\vdash \text{Vbl } b \,.\supset.\, (x)((b \frown invep \frown a) \text{ Den } x \,.\equiv.\, \text{------})$, if '$a$' is taken as the structural description of a sentence '------'.

This clearly holds for 'b' taken as the structural description of a variable. If 'b' is taken as a structural description, but not of a variable, the hypothesis is false. Hence the lemma by *SynR4*. Hence also

$\vdash \text{Tr } a \,.\supset.\, (\mathrm{E}b)(\text{Vbl } b \,.\, (x)((b \frown invep \frown a) \text{ Den } x \,.\equiv.\, \text{------}) \,.\, (x)(b \frown invep \frown a) \text{ Den } x)$,

if 'a' is taken as the structural description of a sentence '------'. Hence

$$\vdash \text{Tr } a \,.\supset.\, (x)\text{------},$$

or, because the quantifier '(x)' here is vacuous,

$$\vdash \text{Tr } a \,.\supset.\, \text{------},$$

where (etc., about 'a'). Conversely, if 'a' is taken as the structural

description of a sentence,

$$\vdash \text{Sent } a,$$

and hence, by *Lemma C2*,

$\vdash$ ------ . Vbl b :$\supset$: Sent a . Vbl b . $(x)((b \frown invep \frown a)$ Den x .$\equiv$. ------) . (x)------,

where (etc., about 'a'). Noting that

$$\vdash (\text{E}b)\text{Vbl } b,$$

and that

$$\vdash \text{Vbl } b \,.\supset.\, \text{PredCon } (b \frown invep \frown a),$$

if (etc., about 'a'), we have that

$$\vdash \text{------} \,.\supset.\, \text{Tr } a,$$

if (etc., about 'a'). This completes the proof of adequacy for the definition (V). That for (IV) is similar.

The proofs of adequacy for the definitions given also establish their mutual equivalence. This may be seen by noting that any sentence which is true according to one definition is also true according to any other and conversely.

For purposes of the subsequent development we shall suppose that 'Tr' is defined by the especially simple definition (V) of § B in terms of vacuous abstracts.

We note now two important consequences of the (Fourth) Adequacy Rule. The first is the *Negation Rule*. It states that if 'a' is taken as the structural description of a sentence, then a is not true if and only if (*tilde a*) is true.

TC4. $\vdash \sim$Tr a .$\equiv$. Tr (*tilde a*), if 'a' is taken as the structural description of a sentence '------' from the translational part of SM_D^L.

The proof is immediate, using the (Fourth) Adequacy Rule, if we note that on the hypothesis '(*tilde a*)' is the structural description of '$\sim$------'. In a similar way one can prove the *Disjunction Rule*,

TC5. $\vdash$ Tr a .v. Tr b :$\equiv$: Tr (*a vee b*), if 'a' and 'b' are taken as structural descriptions of sentences from the translational part of SM_D^L.

Using *SynR4*, we can also prove that

TC6. $\vdash$ Sent a :$\supset$: $\sim$Tr a .$\equiv$. Tr (*tilde a*),

and

TC7. $\vdash$ Sent a . Sent b :$\supset$: Tr (*a vee b*) .$\equiv$. (Tr a .v. Tr b).

D. Correspondence. The condition of adequacy is not a mere idle desideratum. Now that we have shown it to hold, we are assured

that the truth-definitions provide an appropriate kind of correspondence between the expressions of L and the objects of the fundamental domain of L. That this kind of correspondence holds seems essential if we are to use the truth-concept for certain appointed tasks. (See § E and §§ H–I below.) This kind of correspondence is also important on more philosophical grounds. It shows, namely, that the truth-concept as defined here is in essential agreement with age-old concepts concerning the meaning of 'truth'.

When, within the context of a systematic language, we make an assertion that ------, this is equivalent, according to the condition of adequacy, to saying that the expression a is true, where 'a' is taken as the structural description of the sentence '------'. To be sure our ordinary use of the term 'true' as applied to declarative sentences is more or less vague and ill-defined. But it would seem, however, that 'true' as applied to such sentences is usually used in such a way that this equivalence holds. Hence we can think of the precise, modern, semantical concept of truth as giving an explication in Carnap's sense (I,E) of older notions of truth.

Also it seems that the core of what is traditionally called the 'correspondence theory of truth' is provided where the condition of adequacy obtains. The precise meaning of this doctrine is far from clear without a satisfactory account of what is meant by 'correspondence'. Given a language-system L, we can clarify this a little by saying that a sentence a of L is true if and only if what a asserts of the objects of L does in fact obtain. If we now go one step further and strip this last sentence of its inessential words, we end up with the statement of an Adequacy Rule itself. Hence the so-called correspondence theory of truth seems to be explicated by modern semantical definitions of truth for which the condition of adequacy holds. To be sure, 'correspondence' does not appear as a technical term in any way in the precise definitions. But the requirement of adequacy seems to assure that the kind of correspondence intended holds.

Hence, following Tarski, one can regard the modern semantical theory of truth as incorporating the classical, Aristotelian concept of truth.[1] If we should ask *under what conditions* a sentence in an appropriate formalism is true or false, the answer according to the classical conception can be given by the famous dictum of Aristotle:

To say of what is that it is not, or of what is not that it is, is false,

[1] In addition to *Der Wahrheitsbegriff*, see also Tarski's popular paper, "The Semantic Conception of Truth and the Foundations of Semantics", *Philosophy and Phenomenological Research*, 4 (1944), pp. 341–76. Cf. also, e.g., A. Kaplan, "What Good is 'Truth'?", *ibid.*, 15 (1954), pp. 151–70.

while to say of what is that it is or of what is not that it is not, is true.[1]

If the definition of truth is to conform to this conception, all equivalences of the following kind must hold:

The sentence '------' is true if and only if ------.

In other words, the condition of adequacy must obtain.

The condition of adequacy must not, however, be mistaken for a *definition* of truth. In a certain sense, one might wish to regard a specific instance of an Adequacy Rule, taking '*a*' as the structural description of a specific sentence '------', as a partial definition of truth, i.e., as a definition of truth for just the one sentence ***a***. But the full definition of truth must be more general and must be given for *variable* '*a*'. The situation in semantics is similar to that in arithmetic, where, e.g., we might wish to define '+'. We should not be satisfied with definitions of just '(2+3)', '(6+7)', and so on; we should rather require a definition of '$(x+y)$' where 'x' and 'y' are numerical variables. So also in semantics we require a general definition of 'Tr *a*' for variable '*a*'. Just as in arithmetic the general definition of '$(x+y)$' is in effect a logical sum of an infinite number of partial definitions, so also in semantics the general definition of truth gives the effect of an infinite logical conjunction of all partial definitions of the kind mentioned. Just as in arithmetic one must assume recursion equations in some form or presuppose a powerful basic logic, so also in semantics one must assume a powerful logical substructure or achieve the effect of such infinite conjunctions in some other way. Here multiple denotation is taken as a special primitive and enables us to gain what otherwise can be accomplished by employing more powerful logical means.[2] (See (VII,E) below.) Given a general definition of truth, such as one of them above in terms of multiple denotation, one then *proves* a corresponding Adequacy Rule. We are then assured that the definition provides a truth-concept for the language at hand roughly in accord with the classical conception.

It should be emphasized that semantical definitions of truth do not provide a *criterion* by means of which we can determine whether or not a given sentence is true. The semantical definition merely provides an *analysis* of what it means to say that a sentence is true. It gives a necessary and sufficient condition under which we can

[1] Cf. Aristotle, e.g., *Metaphysics*, 1017^a7, 30–5. See also, I. M. Bocheński, *Ancient Formal Logic* (North-Holland Publishing Co., Amsterdam: 1951), p. 31 Also cf. St. Thomas, *De Veritate*, Qu. I, Art. I, Resp.

[2] As, e.g., in *Der Wahrheitsbegriff*.

properly ascribe 'true' to a sentence. To decide, by observation or experiment or by some other means, whether a sentence is in fact true or false is not the affair of semantics. This is, broadly conceived, the task of the special sciences. The distinction between providing a criterion and an analysis is an obvious one, but seems frequently to be overlooked.

It will thus simply not do to dismiss the semantical truth-concept as philosophically irrelevant, as many philosophers have attempted to do. We have in modern semantics a full-fledged, successful explication of the concept of truth, a logical analysis of one of the historically most important philosophical concepts as applied to the sentences of formalized language-systems of certain kinds. Those philosophers who work with formalized language-systems, rather than with perhaps inconsistent natural languages, are merely trying to do more carefully and with more secure intellectual tools what analytically minded philosophers have always been trying to do. In the case of the semantical truth-concept, the analysis or explication seems wholly successful. The age-old problem as to the meaning of the truth-predicate has been wholly clarified, where one restricts attention to formalized language-systems of a certain kind. That such restriction is itself desirable has been urged at some length in Chapter I.

E. Semantical Consistency of L. We have just noted one reason why the semantical concept of truth is so exceedingly important a notion, i.e., that it provides an explication of an important philosophical concept. But there are other reasons also, one of which is of special interest in connection with the notion of *consistency*. In fact, a semantical meta-language containing an adequate truth-concept for L is ordinarily of sufficient power to enable one to prove that L is consistent.

To be somewhat more precise, let us recall from (III,H) that a language-system L, of the kind under consideration, is *consistent* if and only if there exists no formula a of L such that both a and (*tilde* a) are theorems of L. Suppose now that L is known to be inconsistent. We then know that there is at least one formula A of L such that both A and its negative are theorems. But, recalling a law of truth-functional logic, $TB33$(II), that

$$\vdash A \,.\, \sim A :\supset: B,$$

where A and B are any formulae of L, and supposing that

$$\vdash A \text{ and } \vdash \sim A$$

we can prove that

$$\vdash B$$

for any $\boldsymbol{B}$. Hence every formula $\boldsymbol{B}$ of L is a theorem. And, if $\boldsymbol{B}$ is also a sentence it is then presumably true. (See *TE5a* below.) Hence within an inconsistent L all sentences of L are true. Any such L would seem utterly lacking in scientific or philosophical interest. Within an L which is inconsistent in this very basic sense the distinction between truth and falsehood vanishes. There would be no reason to seek a proof of a theorem within an inconsistent L, because one could equally well prove its negative. And similarly, there would be no point for a scientist to attempt to verify by observation or experiment a sentence of an L known by him to be inconsistent, because he would already know it to be true. And he also would know it to be false.

One of the reasons the semantical concept of truth is useful is that it provides the basis for a method of consistency proof of wide applicability. Within a semantical meta-language containing an adequate truth-concept for L, we can ordinarily prove the consistency of L. Much of the mathematical interest in the semantical truth-concept stems from this circumstance.

We now show how, within SM_D^L, one can prove the simple consistency of L.

To prove L consistent we must of course employ an argument. And to carry out an argument is to allow certain methods of deduction. In giving a consistency proof, it is always essential to specify carefully the methods being allowed. Otherwise the "proof" is left hanging at loose ends, and one has no basis on which to judge whether it is sound. The methods of deduction allowed here are those of SM_D^L, not just its logical rules, but its logical and non-logical rules together. For the proof that L is consistent the whole of SM_D^L is required. We thus shall prove the simple consistency of L by means of a semantical argument.

We need a few preliminary theorems concerning the truth-concept. The first states that every (logical or non-logical) axiom of L has a true closure in the sense of 'Clsr''.

TE1a. $\vdash (a)(\mathrm{Ax}\ a\ .\supset .\ (\mathrm{E}b)(b\ \mathrm{Clsr}'\ a\ .\ \mathrm{Tr}\ b))$.

The proof is not difficult. Let 'a' be the structural description of an expression '-----' containing no free variables from the translational part of SM_D^L. Suppose '-----' is an axiom and also a sentence. By the (Fourth) Adequacy Rule,

$$\vdash \mathrm{Tr}\ a\ .\equiv .\ \text{-----},$$

but also

$$\vdash \text{-----}.$$

Thus *TE1a* holds for 'a' so chosen. Next suppose 'a' is taken as an

axiom but not a sentence. Then by a similar argument there is some closure of a which is true. Hence *TE1a* holds for such 'a'. But if 'a' is taken as a structural description, but not the structural description of an axiom, then

$$\vdash \sim \mathrm{Ax}\ a,$$

and hence *TE1a* holds. These together with *SynR4* yield *TE1a*.

Also, using the definition of 'Tr', we have immediately that

TE1b. $\vdash \mathrm{Tr}\ a\ .\supset .\ \mathrm{Sent}\ a.$

The next law states that if a is true and b is a generalization of a, then b is true.

TE2. $\vdash \mathrm{Tr}\ a\ .\ b\ \mathrm{Gen}\ a :\supset: \mathrm{Tr}\ b.$

The proof is not difficult if we recall the law (*TC7a* in II) concerning vacuous quantifiers. Let '-----' be a sentence in the translational part of SM_D^L. We know then in view of *TC7a*(II) that

(3) $$\vdash \text{-----}\ .\supset .\ (x)\text{-----},$$

where x is any translational variable. Taking 'a' as the structural description of '-----' and 'b' as the structural description of '(x)-----', in view of (3) and the Adequacy Rule,

(4) $$\vdash \mathrm{Tr}\ a\ .\ b\ \mathrm{Gen}\ a :\supset: \mathrm{Tr}\ b.$$

Suppose 'b' is now taken as any structural description of an expression which is not a generalization of a. Then

$$\vdash \sim b\ \mathrm{Gen}\ a,$$

and hence (4) holds for all b, by *SynR4*. Finally if we take 'a' as a structural description other than of a sentence, then

$$\vdash \sim \mathrm{Sent}\ a,$$

and hence by *TE1b*,

$$\vdash \sim \mathrm{Tr}\ a.$$

Thus (4) holds for all a by *SynR4*.

We have an analogous law concerning *Modus Ponens*.

TE3. $\vdash \mathrm{Tr}\ a\ .\ \mathrm{Tr}\ b\ .\ c\ \mathrm{MP}\ a,b :\supset: \mathrm{Tr}\ c.$

The proof is similar.

Combining *TE2* and *TE3* we have that

TE4a. $\vdash (c)(d)(e)(\mathrm{Tr}\ d\ .\ \mathrm{Tr}\ e\ .\ c\ \mathrm{IC}\ d,e :\supset: \mathrm{Tr}\ c),$

and as a corollary,

TE4b. $\vdash (\mathrm{E}d)(d\ \mathrm{Clsr}'\ a\ .\ \mathrm{Tr}\ d)\ .\ (\mathrm{E}d)(d\ \mathrm{Clsr}'\ b\ .\ \mathrm{Tr}\ d)\ .\ c\ \mathrm{IC}\ a,b :\supset: (\mathrm{E}d)(d\ \mathrm{Clsr}'\ c\ .\ \mathrm{Tr}\ d).$

The next theorem provides an important lemma for the proof of

consistency of L. It states that all theorems of L which are sentences are true.

TE5. $\vdash (a)(\text{Sent } a \,.\, \text{Thm } a :\supset: \text{Tr } a)$.

The proof is by the Rule of Framed Ingredients, *TG4* (of III). For by it,

$\vdash a \text{ FrIng } b \,.\, (c)(c \text{ FrIng } b :\supset: (\text{E}a')(a' \text{ Clsr}' \, c \,.\, \text{Ax } c) \,.\text{v}.\, (\text{E}d)(\text{E}e)(d \text{ Pr}_b \, c \,.\, e \text{ Pr}_b \, c \,.\, c \text{ IC } d,e)) \,.\, (c)(\text{Ax } c \,.\supset.\, (\text{E}a')(a' \text{ Clsr}' \, c \,.\, \text{Tr } c)) \,.\, (c)(d)(e)((\text{E}a')(a' \text{ Clsr}' \, d \,.\, \text{Tr } a') \,.\, (\text{E}a')(a' \text{ Clsr}' \, e \,.\, \text{Tr } a') \,.\, c \text{ IC } d,e :\supset: (\text{E}a')(a' \text{ Clsr}' \, c \,.\, \text{Tr } a')) :\supset: (\text{E}a')(a' \text{ Clsr}' \, a \,.\, \text{Tr } a')$.

But the third and fourth hypotheses here are stipulated as theorems by *TE1a* and *TE4b* above. Hence these hypotheses may be dropped. By appropriate generalization and use of the definition of 'Thm' and using *TG12c*(III), we gain *TE5*. (Note that *TE5* could also be proved in the manner of *TE1a*. But cf. *TD10*(IX) below.)

One more lemma is needed.

TE6. $\vdash \text{Sent } a \,.\supset.\, \sim(\text{Tr } a \,.\, \text{Tr } (\textit{tilde } a))$.

The proof is immediate by *TC6* above.

The *First Consistency Rule* is as follows.

TE7. $\vdash \sim(\text{E}a)(\text{Sent } a \,.\, \text{Thm } a \,.\, \text{Thm } (\textit{tilde } a))$.

For the proof we note that

$$\vdash \text{Sent } a \,.\, \text{Thm } (\textit{tilde } a) :\supset: \text{Tr } (\textit{tilde } a),$$

by *TE5* and *TG8e*(III). Using again *TE5*, we have that

$$\vdash \text{Sent } a \,.\, (\text{Thm } a \,.\, \text{Thm } (\textit{tilde } a)) :\supset: (\text{Tr } a \,.\, \text{Tr } (\textit{tilde } a)),$$

and hence that

$$\vdash \text{Sent } a \,.\, \sim(\text{Tr } a \,.\, \text{Tr } (\textit{tilde } a)) :\supset: \sim(\text{Thm } a \,.\, \text{Thm } (\textit{tilde } a)).$$

From this we get *TE7* by *TE6*.

A *Second Consistency Rule* can be proved to the effect that there is no *formula*, whether a sentence or not, which is such that both it and its negation are theorems.

TE8. $\vdash \sim(\text{E}a)(\text{Fmla } a \,.\, \text{Thm } a \,.\, \text{Thm } (\textit{tilde } a))$.

We can prove this indirectly by supposing that there is such a formula. But from syntax,

$$\vdash \text{Thm } a \,.\, \text{Thm } (\textit{tilde } a) :\supset: \text{Thm } (a \textit{ dot tilde } a),$$
$$\vdash \text{Fmla } a \,.\, \text{Thm } (a \textit{ dot tilde } a) \,.\, \text{Sent } b :\supset: \text{Thm } b,$$

and

$$\vdash \text{Fmla } a \,.\, \text{Thm } (a \textit{ dot tilde } a) \,.\, \text{Sent } b :\supset: \text{Thm } (\textit{tilde } b).$$

Hence

$\vdash$ Fmla a . Thm a . Thm (*tilde a*) . (Eb)Sent b :$\supset$: (Eb)(Sent b . Thm b . Thm (*tilde b*)).

By contraposition and *TE7*, we get *TE8*, noting that

$$\vdash \text{(E}b\text{)Sent } b.$$

These consistency proofs show that the consistency of L can be established within SM_D^L. And note that we can establish this whether in fact L is consistent or not. For if L is inconsistent then SM_D^L, containing a translation of L, is also inconsistent and hence contains all its formulae as theorems. But if L is consistent and moreover consistent jointly with syntax, then SM_D^L is also consistent as we shall see in a moment. (Cf. also (XIII,E).)

F. Some Further Notions of Semantics. In terms of the semantical truth-concept for L, several other important semantical notions are definable.[1] Some of these are interesting for their own sake, or correspond more or less roughly with important concepts in current use.

Clearly a sentence is *false* if and only if it is not true.

'Fls a' abbreviates '(Sent a . $\sim$Tr a)'.

A sentence a is said to *implicate* (*materially*) a sentence b if and only if a is false or b is true.

'a Imp b' abbreviates '(Sent a . Sent b : Fls a .v. Tr b)'.

A sentence a is *equivalent* to a sentence b if and only if a and b are both true or both false.

'a Equiv b' abbreviates '(Tr a . Tr b :v: Fls a . Fls b)'.

Sentences a and b are said to be *disjunct* if at least one of them is true,

'a Disj b' abbreviates '(Sent a . Sent b : Tr a .v. Tr b)',

and are said to be *exclusive* if not both are true,

'a Excl b' abbreviates '(Sent a . Sent b . $\sim$(Tr a . Tr b))'.

Finally, sentences a and b are said to be *conjuncts* if both are true, i.e., if they are not exclusive,

'a Conj b' abbreviates '(Sent a . Sent b . $\sim a$ Excl b)'.

With the help of these definitions we can construct precise semantical analogues of the so-called *truth-tables*. First let us consider negation. Clearly

TF1a. $\vdash$ Sent a :$\supset$: Tr a .$\equiv$. $\sim$Fls a,

[1] Cf. R. Carnap, *Introduction to Semantics*, pp. 35–40 and *passim*.

and of course

TF1b. $\vdash$ Sent a :$\supset$: Fls a .$\equiv$. $\sim$Tr a.

Concerning disjunction, we can prove that

TF2a. $\vdash$ Sent a . Sent b :$\supset$: a Disj b .$\equiv$. (Tr a . Tr b :v: Tr a . Fls b :v: Fls a . Tr b),

and

TF2b. $\vdash$ Sent a . Sent b :$\supset$: $\sim a$ Disj b .$\equiv$. (Fls a . Fls b).

Note also the following theorems concerning equivalence.

TF3a. $\vdash$ Sent a . Sent b :$\supset$: a Equiv b .$\equiv$. (Tr a . Tr b :v: Fls a . Fls b).

TF3b. $\vdash$ Sent a . Sent b :$\supset$:$\sim a$ Equiv b .$\equiv$. (Tr a . Fls b :v: Fls a . Tr b).

In the case of the relation Imp of material implication, we have that

TF4a. $\vdash$ Sent a . Sent b :$\supset$: a Imp b .$\equiv$. (Tr a . Tr b :v: Fls a . Tr b :v: Fls a . Fls b),

and

TF4b. $\vdash$ Sent a . Sent b :$\supset$: $\sim a$ Imp b .$\equiv$. (Tr a . Fls b).

And, finally, in the case of conjunction,

TF5a. $\vdash$ Sent a . Sent b :$\supset$: a Conj b .$\equiv$. (Tr a . Tr b),

and

TF5b. $\vdash$ Sent a . Sent b :$\supset$: $\sim a$ Conj b .$\equiv$. (Tr a . Fls b :v: Fls a . Tr b :v: Fls a . Fls b).

As consequences of *TC6* and *TC7*, we have also that

TF6a. $\vdash$ Sent a . Sent b :$\supset$: Tr (a *hrsh* b) .$\equiv$. a Imp b,
TF6b. $\vdash$ Sent a . Sent b :$\supset$: Tr (a *dot* b) .$\equiv$. a Conj b,
TF6c. $\vdash$ Sent a . Sent b :$\supset$: Tr (a *tripbar* b) .$\equiv$. a Equiv b,
TF7a. $\vdash$ Sent a . Sent b :$\supset$: a Imp b .$\equiv$. (Tr a .$\supset$. Tr b),

and

TF7b. $\vdash$ Sent a . Sent b :$\supset$: a Equiv b .$\equiv$. (Tr a .$\equiv$. Tr b).

By *DenR2*, it is evident that such logical constants as 'v', '$\sim$', and so on, of L cannot be regarded as denoting in the sense of 'Den', not being predicate constants. But nonetheless we can prove the salient semantical properties of these notions, as the theorems above indicate. More particularly, *TC6* and *TC7* above give the basic semantical properties of '$\sim$' and 'v' respectively. And the theorems just cited give the basic properties of '.', '$\supset$', and '$\equiv$'. These notions are in effect interpreted in SM^L_D, in the sense that the truth

(or falsity) of statements of L containing them are reducible to the truth (or falsity) of the constituents of these statements. Thus for present purposes there is no need to regard '$\sim$' or 'v' as denoting in any sense.

In (III,H) the notion of completeness of a virtual class of sentences was defined. The notion there was syntactical, no reference to semantical notions being made in the definiens. We may now, in terms of the truth-concept, define a *semantical* notion of completeness as follows.

'SemCmplt $\boldsymbol{F}$' abbreviates '$((a)(\boldsymbol{F}a \mathrel{.\supset} . \text{Sent } a) \mathrel{.} (a)(\text{Tr } a \mathrel{.} \boldsymbol{F}a \mathrel{:\supset:} \text{Thm } a))$'.

A virtual class of sentences is complete in this sense provided every true sentence which is a member of it is a theorem of L.

These two notions of completeness for L are equivalent on a suitable hypothesis about $\boldsymbol{F}$.

TF8. $\vdash (a)(\boldsymbol{F}a \mathrel{.\supset} . \boldsymbol{F}(\textit{tilde } a)) \mathrel{:\supset:} \text{Cmplt } \boldsymbol{F} . \equiv . \text{SemCmplt } \boldsymbol{F}$.

The proof from right to left is by *TC6*. From left to right, we note that, by *TE5*,

$\vdash (a)(\boldsymbol{F}a \mathrel{:\supset:} \text{Sent } a \mathrel{.} (\sim\text{Thm } a \mathrel{.\supset} . \text{Thm } (\textit{tilde } a))) \mathrel{.\supset} . (a)(\boldsymbol{F}a \mathrel{.\supset} . \text{Sent } a) \mathrel{.} (a)(\boldsymbol{F}a \mathrel{.} \sim\text{Tr } (\textit{tilde } a) \mathrel{:\supset:} \text{Thm } a)$.

Hence the theorem, using again *TC6*.

If we take $\boldsymbol{F}$ here as '$a \supset \text{Sent } a$' we gain a theorem which says in effect that L is complete in one of these senses if and only if it is also complete in the other.

Within L, and hence within SM^L_D, we can gain the effect of saying that the fundamental domain of L consists of just one object, of two objects, and so on for any fixed finite n. We let

'FD$L1$' abbreviate '$(x)(y)y = x$',
'FD$L2$' abbreviate '$(\text{E}x)(\text{E}y)(\sim x = y \mathrel{.} (z)(z = x \mathrel{.\text{v}.} z = y))$',

and so on. There exist systems L whose fundamental domains consist of just a fixed finite number n of objects, and hence these notions may be useful in describing some of their semantical properties.

We note finally (and this will be useful later) that there is an L which is complete, consistent, and whose fundamental domain consists of just *one* object. To see this, let L contain just one non-logical primitive predicate 'P', of one place, and two non-logical rules, that

$$\vdash \text{P}x,$$

and

$$\vdash x = y.$$

Let this system be called L^*. Within $SM_D^{L^*}$, we wish to show that

$$\vdash \text{Cnst } a \supset \text{Sent } a \,.\, \text{Cmplt } a \supset \text{Sent } a \,.\, \text{FD}L^*1.$$

That

$$\vdash \text{Cnst } a \supset \text{Sent } a,$$

follows from *TE7*. And that 'FDL^*1' holds is immediate. But also we can show that *every* formula of L^* is a theorem and hence that L^* is complete. To see this we note the following syntactical theorems.

TF9a. $\vdash (c)(d)(e)((\text{Thm } d \,.\text{v.}\, \text{Thm } (\textit{tilde } d)) \,.\, (\text{Thm } e \,.\text{v.}\, \text{Thm } (\textit{tilde } e)) \,.\, c = (d \textit{ vee } e) :\supset: \text{Thm } c \,.\text{v.}\, \text{Thm } (\textit{tilde } c))$.

TF9b. $\vdash (c)(d)((\text{Thm } d \,.\text{v.}\, \text{Thm } (\textit{tilde } d)) \,.\, c = (\textit{tilde } d) :\supset: \text{Thm } c \,.\text{v.}\, \text{Thm } (\textit{tilde } c))$.

TF9c. $\vdash (c)(d)((\text{Thm } d \,.\text{v.}\, \text{Thm } (\textit{tilde } d)) \,.\, (c \text{ Gen } d \,.\text{v.}\, c \text{ Abst } d) :\supset: \text{Thm } c \,.\text{v.}\, \text{Thm } (\textit{tilde } c))$.

Hence

TF10. $\vdash (c)(d)(e)((\text{Thm } d \,.\text{v.}\, \text{Thm } (\textit{tilde } d)) \,.\, (\text{Thm } e \,.\text{v.}\, \text{Thm } (\textit{tilde } e)) \,.\, (c = (\textit{tilde } d) \,.\text{v.}\, c = (d \textit{ vee } e) \,.\text{v.}\, c \text{ Gen } d \,.\text{v.}\, c \text{ Abst } d) :\supset: \text{Thm } c \,.\text{v.}\, \text{Thm } (\textit{tilde } c))$.

(*TF9a* through *TF10* hold for any L). But for L^* also

TF11a. $\vdash (a)(\text{AtFmla } a \,.\supset.\, \text{Thm } a)$.

Note that the two axioms of L^* are both atomic formulae. But by generalizing these and then using *R5'* we can prove as a theorem any other atomic formula. As an immediate consequence of *TF11a* we have that

TF11b. $\vdash (a)(\text{AtFmla } a :\supset: \text{Thm } a \,.\text{v.}\, \text{Thm } (\textit{tilde } a))$.

Using now *TF10* and the Rule of Framed Ingredients *TG4*(III), we have that

TF12a. $\vdash (a)(\text{Fmla } a :\supset: \text{Thm } a \,.\text{v.}\, \text{Thm } (\textit{tilde } a))$,

and hence also that

TF12b. $\vdash (a)(\text{Sent } a :\supset: \text{Thm } a \,.\text{v.}\, \text{Thm } (\textit{tilde } a))$.

Thus we know that L^* is complete and hence that

TF13. $\vdash \text{Cnst } a \supset \text{Sent } a \,.\, \text{Cmplt } a \supset \text{Sent } a \,.\, \text{FD}L^*1$.

TF13 is not especially important for its own sake, but will be useful in (IX,G) in the proof of the relative consistency of non-translational semantics.

G. Relative Consistency of SM_D^L. It has been shown above that within SM_D^L we can establish the consistency of L. But what assurance do we have that the rules and axioms of SM_D^L will not them-

selves lead into contradiction? For a full answer to this question we should need, of course, a consistency proof for the whole of SM_D^L. Such a proof will not be given here. The character of such a proof would depend fundamentally on the axiomatic structure of L. But if we suppose for the moment that L together with elementary syntax are jointly consistent (in an appropriate sense), we can then show that SM_D^L is also consistent as follows.[1]

Suppose that the language-system we get by joining a translation of L to M is consistent. From this joint language we form SM_D^L by taking an additional primitive 'Den'. The two Rules adopted concerning 'Den' are *DenR1* and *DenR2* (IV,D). Recall that *DenR1* is a kind of reducibility law. All questions as to whether certain expressions of L denote such and such objects are to be settled in SM_D^L primarily by appeal to this law. Questions concerning denotation are reduced in effect to non-linguistic questions. Any contradiction to be gained from *DenR1* reduces immediately to a contradiction in L.

Let A be a formula of SM_D^L in primitive notation containing one or more occurrences of 'Den' in contexts of the form 'a Den x', where 'a' is taken as the structural description of 'P' or of an abstract '$x\ni$---x---' containing no free variables. Suppose we wish to prove A within SM_D^L. We should proceed as follows. We first contruct a corresponding formula A' of SM_D^L containing no occurrences of 'Den', by replacing all parts 'a Den x' of A by '---x---' if 'a' is taken as '$x\ni$---x---' or by 'Px' if 'a' is taken as '*pee*'. If $\vdash A'$ of SM_D^L without using *DenR1*, we can then prove that $\vdash A$. This we do merely by noting that

$$(1) \qquad \vdash A' \equiv A,$$

assuming *DenR1*. If A were found to lead to a contradiction, A' would also. But if A' is provable without using *DenR1*, it cannot lead to a contradiction on the assumption that the translational and syntactical parts of SM_D^L are jointly consistent. Hence the particular instances of *DenR1* used in proving (1) are consistent with the preceding axioms of SM_D^L, and if we add them to the rules of SM_D^L the only contradiction we could get would be one which could be proved from the preceding rules.

Any other finite number of instances of *DenR1* can be shown relatively consistent in the same way. But all of the instances of *DenR1* together form an infinite totality. Suppose for a moment we could prove a contradiction from this infinite totality. In such a proof we either use *SynR4* or not. Suppose we do not. Any such

[1] Cf. Tarski, *Der Wahrheitsbegriff*, pp. 379–86.

contradiction can then be transformed into a contradiction provable from a finite class of instances of *DenR1*, because the proof in this case can contain only a finite number of steps and hence only a finite number of instances of *DenR1* can be used. But any finite class of instances of *DenR1* is relatively consistent in the way shown. Therefore no contradiction can be proved from the totality of instances of *DenR1* if *SynR4* is *not* used. Now, suppose *SynR4 is* used in a supposed proof of a contradiction, together with the totality of instances of *DenR1*. We may then suppose the replacement procedure of the preceding paragraph carried out, deriving the contradiction therewith without using *DenR1*. Therefore no contradiction can be gained from the totality of instances of *DenR1* using *SynR4* if the underlying syntax and object-language are jointly consistent.

If we now add *DenR2* to the preceding rules of SM^L_D, a similar result follows. From *DenR2* we can immediately prove that

(2) $\vdash a \text{ Den } x \,.\supset.\, \text{PredCon } a,$

if 'a' is taken as the structural description of 'P' or of an abstract (containing no free variables), and also

(3) $\vdash a \text{ Den } x \,.\supset.\, \text{PredCon } a,$

if 'a' is taken as the structural description of an expression of L which is *not* the structural description of 'P' or of an abstract. (Also from (2) and (3) together, using *SynR4*, we can get *DenR2*, so that (2) and (3) together are deductively equivalent with *DenR2*.) Now consider any theorem $\boldsymbol{A}$ of SM^L_D containing 'Den' and provable without using *DenR2*. $\boldsymbol{A}$ cannot be contradictory, unless L with M jointly already contains a contradiction. First let $\boldsymbol{A}$ contain 'Den' only in contexts 'a Den $\boldsymbol{x}$' (where 'a' is taken as the structural description of 'P' or of an abstract, and $\boldsymbol{x}$ is any variable from the translational part of SM^L_D). In this case a is already a predicate constant so that nothing further can be proved by *DenR2* (more specifically, by (2)). In particular, then, no contradiction can be proved using *DenR2*. Consider now an $\boldsymbol{A}$, provable without using *DenR2*, containing a part of the form 'a Den $\boldsymbol{x}$' where 'a' is taken as a structural description but not the structural description of 'P' or of an abstract. No such $\boldsymbol{A}$ is contradictory by hypothesis. For any such 'a', we know that

$$\vdash \sim\text{PredCon } a,$$

and hence by *DenR2* (or (3) above),

$$\vdash (\boldsymbol{x})\sim a \text{ Den } \boldsymbol{x}.$$

Now let $\boldsymbol{A}'$ be formed from $\boldsymbol{A}$ by replacing all such parts 'a Den $\boldsymbol{x}$' by '$(\text{P}\boldsymbol{x} \,.\, \sim\text{P}\boldsymbol{x})$'. Clearly then

$$\vdash a \text{ Den } x .\equiv. (\text{P}x . \sim\text{P}x),$$

on the hypothesis about 'a'. Therefore

$$\vdash A \equiv A',$$

and any contradiction proved by the use of *DenR2* is reducible to one proved without it. Hence no contradiction can arise by the addition of *DenR2* to the other rules.

This argument is still somewhat vague and is couched informally in a natural language. To make the argument more precise, we should need to employ a formalized meta-meta-language. There is no need of constructing such a language for present purposes, however, and this informal argument must suffice to convince us of the relative consistency of SM_D^L.

H. Self-Reference. We know from the preceding section that if L and M are jointly consistent, SM_D^L is consistent also, and hence the semantical paradoxes, in particular that of Epimenides discussed very roughly in (I,F), cannot arise. We are assured, therefore, that SM_D^L provides a secure basis for the semantics of L.

In constructing SM_D^L the distinction between object- and meta-language has been meticulously maintained. The question naturally arises, however, as to whether for security (i.e., consistency) this distinction *must* be maintained or whether it can perhaps be weakened in some respects or even relaxed altogether. In other words, we may ask whether or to what extent a language L can contain its own meta-language. Because the meta-language may be either syntactical or semantical, we have two kinds of questions to consider. Thus we come to the important topics of *syntactical* and *semantical self-reference.*

It has been shown by Gödel that any language-system L (of the familiar classical kinds), containing elementary arithmetic, does in a sense contain its own syntax.[1] The method consists essentially in setting up a correlation between numbers and expressions of L in such a way that all the syntactical notions defined in Chapter III above and *passim* become arithmetical notions. Very roughly, the method is as follows.

Of course L must contain arithmetic or some equivalent. Suppose it to contain arithmetic, and hence the numerals '1', '2', etc., either as defined or primitive constants. Let the primitive symbols of L be given some arbitrary order. Then we can think of the number 1

[1] See especially K. Gödel, "Über Formal Unentscheidbare Sätze der *Principia Mathematica* und Verwandter Systeme I", *Monatshefte für Mathematik und Physik*, 38 (1931), pp. 173–98.

being correlated with the first symbol, 2 with the second, and so on. Rather than to speak of the primitive symbols of L within M, we can now speak instead of their correlated numbers within L itself. Also corresponding to concatenation, we have now a certain numerical operation. We may therefore think of each of the syntactical predicates of Chapter III as now being, in virtue of the correlation, arithmetical predicates. Using 'x', 'x''', etc., as variables of L, we can define

'x Ends x''',

e.g., to mean that the expression whose correlated Gödel number is x ends (in the sense defined in (III,D)) the expression whose Gödel number is x'. And similarly for the other notions. In this way syntax becomes a branch of arithmetic. No inconsistency seems to arise within L because of this correlation.

Let us consider very briefly an important circumstance which results if L contains its own arithmetized syntax in the way just described.

We assume that arithmetical correlates of *SynR1–SynR3*, *TG4*(III), and *TG13*(III) are provable in L. This is desirable if L is to contain its full syntax in the sense discussed in Chapter III. We need not assume that an arithmetical correlate of *SynR4* is provable, because the notion of *proof* for a system containing such a rule allows an infinity of premisses, as we have already noted. The notion of proof in this extended sense has not been defined here.

Let the various formulae of L be enumerated in some way so that we can distinguish a first, a second, and so on. Let f_0x be the Gödel number assigned to the x-th formula in this enumeration. f_0 is a one-place numerical function having integers as arguments and integers as functional values, and is definable within L. (The actual definition presupposes some arithmetical technicalities and need not be given here.) In terms of 'f_0' another numerical function is definable as follows.

'g_0y' abbreviates '(*ex exisqu* (*ex*$\frown$*id*$\frown$*y dot* f_0y))'.

'*ex*', '*exisqu*', '$\frown$', etc., are used here in their arithmetized sense. Given some integer y, g_0y is the Gödel number of the formula

'(Ex)($x = y$. ---x---)',

where '---x---' is the y-th formula in the enumeration and is a sentential function of the one variable 'x'. All the notions involved here are definable within L, if we assume it to contain a sufficiently powerful arithmetic.

Suppose now that, in addition to containing its own syntax, L

also contains its own semantical truth-concept. Thus, suppose 'Tr' is either a defined or primitive numerical predicate of L expressing the notion of truth in L. We also assume that 'Tr' is introduced in such a way as to be an adequate predicate in the sense defined above. Consider now the statement that

(1) $$\vdash (x)(\text{---}x\text{---} .\equiv. \sim\mathrm{Tr}\, g_0 x),$$

where '---x---' is some sentential function of the one variable 'x'. L contains such formulae as theorems in view of the correlation between numerical and syntactical notions, and because 'Tr' is assumed to be introduced arithmetically. The sentential function

(2) '---x---'

itself occurs in the enumeration f_0. Suppose it has the number k_0, so that $f_0 k_0$ is the Gödel number of the formula (2). From (1) we have, by a quantificational rule, that

(3) $$\vdash \text{---}k_0\text{---} .\equiv. \sim\mathrm{Tr}\, g_0 k_0.$$

But, using now an Adequacy Rule, we have also that

$$\vdash A .\equiv. \mathrm{Tr}\, g_0 k_0,$$

where A is the formula of L of which $g_0 k_0$ is the Gödel number. This formula is clearly

$$\text{'}(\mathrm{E}x)(x = k_0 . \text{---}x\text{---})\text{'}.$$

Hence we have that

(4) $$\vdash (\mathrm{E}x)(x = k_0 . \text{---}x\text{---}) .\equiv. \mathrm{Tr}\, g_0 k_0.$$

From (3) and (4) we can immediately derive a contradiction, using an arithmetical analogue of *TE8*(II).

It was suggested above that, so far as is known, a *syntactically* self-referent language L is harmless. Unless there is some prior reason for thinking such an L inconsistent, there is no reason for thinking it so now in view of the syntactical self-reference. From the argument just given, however, *semantical* self-reference is dangerous and may lead to antinomy. The sentence asserted to be a theorem by (3) states that a certain sentence

(5) '---k_0---'

holds if and only if it is not true. The very sentence (5) talks about itself, ascribing to itself in effect the property of falsity. Self-referential sentences of this kind are dangerous and lead directly under suitable circumstances into contradiction.

We have shown, following Tarski, that any language-system L (of the kind under consideration) can be shown inconsistent if it contains its own adequate truth-concept, and the proof is essentially an

adaptation of the kind of argument used in the derivation of the Epimenides paradox.[1] Self-reference of a semantical kind is in some sense too much and leads to contradiction. Some kinds of self-reference are therefore legitimate, whereas other kinds are intolerable. Precisely how one draws the dividing line between tolerable and intolerable kinds of self-reference is by no means clear at the present time.[2]

Nothing that has been said here rules out the possibility that a semantical meta-language of another kind could be constructed in which the semantical paradoxes would not arise but which could be interpreted in such a way as to contain in some sense its own truth-concept. Such a meta-language we shall in fact formulate in a later chapter (X,G), in connection with non-translational semantics.

We note that in order for SM_D^L to contain consistently an adequate truth-concept for L, SM_D^L must be *essentially richer* in modes of expression and deduction than L itself.[3] It must contain, as we have already noted, not only translations of the symbols of L, but also structural-descriptive names of those symbols, together with 'Den'. Over and above these new primitives, SM_D^L must contain additional axioms, as we have seen. These are provided by *SynR1–SynR4* and by *DenR1–DenR2*, which provide greater deductive power than that of L alone.

If L contains arithmetic, we can gain the effect of the structural-descriptive names via Gödel's method of arithmetization within L itself. But we could not also gain the effect of 'Den' within L and retain consistency, because if 'Den' or an arithmetical equivalent were available, we could define 'Tr' and hence reconstruct the Epimenides paradox. But we could develop consistently the semantics of L by adding a semantical primitive in terms of which an arithmetical predicate 'Tr' is definable. Suppose we take, as above, a form of multiple denotation as primitive. *DenR1* and *DenR2* become now arithmetical rules characterizing this primitive. And *SynR1–SynR4* become arithmetical axioms or theorems of L, *SynR4* becoming a rule of infinite induction for integers. The resulting meta-language, which we may call SM_D^{L+}, is in obvious ways essentially richer than L. The consistency of SM_D^{L+} relative to that of L may be proved essentially as in § G. SM_D^{L+} is an especially simple meta-

[1] Cf. A. Tarski, *Der Wahrheitsbegriff*, pp. 370–8.

[2] Cf. F. B. Fitch, *Symbolic Logic An Introduction*, pp. 217–25.

[3] Cf. A. Tarski, *Der Wahrheitsbegriff*, *passim*, and "Grundlegung der Wissenschaftlichen Semantik", *Actes du Congrès International de Philosophie Scientifique*, III (1936), pp. 1–8.

language, for L's containing arithmetic, and is of some interest on its own account.

If L is consistent it cannot contain its own adequate truth-concept. Nor can SM_D^L or SM_D^{L+} contain *their* own adequate truth-concepts respectively. But if L is taken as an $SM_D^{L'}$, for an appropriate L', a semantical meta-language based on multiple denotation is forthcoming by a suitable extension of the methods described. (Because L in this case contains *SynR4*, the notion of 'Thm' must be suitably extended or taken as an additional syntactical primitive.[1]) This meta-language may be called

$$SM_D^{SM_D^{L'}}.$$

It could contain consistently an adequate truth-concept, not for itself, but for $SM_D^{L'}$. L' could in turn be taken as a semantical meta-language of some L'', and so on.

I. Undecidability and Truth. In closing the discussion in this chapter of the semantical concept of truth, we note very roughly its connection with the important notion of *undecidability*. A sentence of L is said to be an *undecidable sentence* of L if and only if neither it nor its negative is a theorem of L.

'US a' abbreviates '(Sent a . ~Thm a . ~Thm (*tilde* a))'.

The existence of sentences which are undecidable may be proved very briefly as follows. For this proof we can use SM_D^{L+}. We may suppose this meta-language developed along lines similar to that of SM_D^L. Hence the various theorems of SM_D^L given above now reappear as arithmetical theorems of SM_D^{L+}. (The syntactical variables 'a', etc., are now to be construed as numerical variables.) We shall refer to these arithmetical theorems by number, by referring instead to the corresponding theorems of SM_D^L.

Within SM_D^{L+} we can prove, using the argument of Gödel, a formula closely akin to (1) in § H above, but with 'Thm' replacing 'Tr'.

(1) $\vdash (a)(\text{---}a\text{---} .\equiv. \sim\text{Thm } g_0 a),$

taking 'g_0' as in § H. From this, as in § H, where '---a---' is the k_0-th formula in the enumeration f_0, we have also that

(2) $\vdash \text{---}k_0\text{---} .\equiv. \sim\text{Thm } g_0 k_0.$

[1] For the syntactical properties of 'Thm' in this extended sense, see, e.g., B. Rosser, "Gödel Theorems for Non-Constructive Logics", *The Journal of Symbolic Logic*, 2 (1937), pp. 129–37.

From the Adequacy Rules (§ C) and *TE8*(II), we have that

$$\vdash \text{---}k_0\text{---} .\equiv. \text{Tr } g_0k_0.$$

Let $j_0 = g_0k_0$. Then

(3) $$\vdash \text{Tr } j_0 .\equiv. \sim\text{Thm } j_0.$$

The sentence '---k_0---' has the extraordinary property of being true if and only if it is not a theorem!

We can now see that neither the sentence '---k_0---' nor its negative is a theorem of L. From *TE6*, we know that

(4) $$\vdash \text{Sent } a . \supset . \sim(\text{Tr } a . \text{Tr } (\textit{tilde } a)),$$

construing (*tilde a*) arithmetically. (Where a is the Gödel number of a formula, (*tilde a*) is the Gödel number of the negation of that formula.) By *TE5a* we have that

(5) $$\vdash \text{Sent } a :\supset: \text{Thm } a .\supset. \text{Tr } a,$$

and also, similarly, that

(6) $$\vdash \text{Sent } a :\supset: \text{Thm } (\textit{tilde } a) .\supset. \text{Tr } (\textit{tilde } a).$$

Using (3) and (5) and recalling the theorems of truth-functional logic that

$$\vdash \sim A \supset A . \supset . A$$

and

$$\vdash A \equiv \sim B .\equiv. \sim A \equiv B,$$

we have that

(7) $$\vdash \text{Sent } j_0 .\supset. \text{Tr } j_0,$$

and hence by (3) again that

$$\vdash \text{Sent } j_0 .\supset. \sim\text{Thm } j_0.$$

Similarly by (4), (6), and (7),

$$\vdash \text{Sent } j_0 .\supset. \sim\text{Thm } (\textit{tilde } j_0).$$

Hence, by combining these and dropping 'Sent j_0', which clearly holds,

(8) $$\vdash \sim\text{Thm } j_0 . \sim\text{Thm } (\textit{tilde } j_0).$$

Any such sentence j_0 is undecidable in L, because neither it nor its negative is a theorem of L.

But note also that in view of (7), j_0 is true. Within SM_D^{L+}, a formalism containing an adequate truth-concept for L, the undecidable sentences of L become true. We see then that SM_D^{L+} provides a systematic method of handling the undecidable sentences of L.[1]

Note that this proof of the existence of undecidable sentences

[1] This argument is essentially that of Tarski, *Der Wahrheitsbegriff*, pp. 401–2.

involves essential reference to the truth-concept, and hence is given in the *semantical* meta-language SM_D^{L+}. The existence of such sentences can also be proved within L itself if L contains its own syntax, on the hypothesis that L is consistent. We have as a theorem then that

(9) $\quad \text{Cnst } a \ni \text{Fmla } a :\supset: \sim\text{Thm } j_0 \,.\, \sim\text{Thm } (\textit{tilde } j_0).$

The proof of this involves some arithmetical technicalities which need not concern us here.[1]

An important corollary of (9) should be mentioned, namely, that if L is consistent then the statement of L which says that L is consistent is not itself a theorem of L. Let m_0 be the Gödel number of this statement, i.e., the Gödel number of 'Cnst $a \ni$ Fmla a'. Then,

(10) $\quad \text{Cnst } a \ni \text{Fmla } a \,.\supset.\, \sim\text{Thm } m_0.$

For, suppose that

$$\text{Cnst } a \ni \text{Fmla } a \,.\, \text{Thm } m_0.$$

Then, in view of (9), we should know that

$$\sim\text{Thm } j_0.$$

Also, recalling that j_0 is the Gödel number of '$\sim j_0$' or '---k_0---', we know, also from (9), that

$$\text{Thm } (m_0 \textit{ hrsh } j_0),$$

taking '*hrsh*' now in its arithmetized sense. Hence

$$\sim\text{Thm } j_0 \,.\, \text{Thm } (m_0 \textit{ hrsh } j_0).$$

But then, using the arithmetical analogue of *TG7b*(III), we should have that

$$\sim\text{Thm } j_0 \,.\, \text{Thm } j_0,$$

which is contradictory. Therefore (10) holds.

In discussing self-reference and undecidability we have employed an arithmetized syntax. There are also other methods which do not explicitly use arithmetic, but which are not essentially different. (We could, e.g., following Quine, show that M or L contains undecidable sentences by appropriately interpreting '$\frown$' and the structural-descriptive primitives. Cf. (X,G).) But we have preferred to follow Gödel's original method, even though the account of it is perforce vague and inexact in some respects.

To summarize. We have seen in this chapter that the semantical concept of truth is of vital importance for five reasons. (1) It provides an explication of age-old philosophical notions concerning

[1] See Gödel, *op. cit.* Also B. Rosser, "Extensions of Theorems of Gödel and Church", *The Journal of Symbolic Logic*, 1 (1936), pp. 87–91.

truth. (2) The meta-languages containing it provide a method of consistency proof of wide generality. (3) The truth-concept provides a basis in terms of which other important semantical notions may be defined. (4) Considerations concerning the truth-concept enable us to comprehend more fully the nature of semantical self-reference. And (5) the meta-languages containing the truth-concept of L provide a systematic way of handling any undecidable sentences which L may contain.

CHAPTER VI

SET THEORY AND THEORY OF TYPES

THE semantics developed in the preceding chapters is applicable to any first-order object-language L containing no individual or functional constants as primitive. The syntactical description of the logical part of L, so to speak, was formulated explicitly in Chapter III. But the non-logical part was left there at loose ends. It was specified that, in addition to its logical axioms, L must contain some non-logical or descriptive axioms characterizing the properties or relations referred to by the primitive predicate constants. Also it was specified that translations of these non-logical axioms must reappear as axioms of the semantical meta-language SM_D^L. Let us now consider the non-logical axioms of certain L somewhat more closely, by giving an exact syntactical and semantical analysis of them.

If we take L as the Hilbert–Bernays system (Z) sketched in (II,L), the non-logical axioms are the axioms $Z1$–$Z7$ as given there. If we take L as a language expressing certain portions of classical physics, say Newtonian mechanics, the non-logical axioms would then comprise in part certain basic laws of that science. If we take L as a language-system for the classical theory of real numbers, the non-logical axioms could be taken, e.g., as axioms studied by Huntington or Tarski.[1] For the study of the non-logical part of L, we might investigate any one of these languages in greater detail. Rather than any one of these, however, we shall study in this chapter two important first-order language-systems of a much more powerful kind. These will be (1) a system based upon the *Zermelo set theory*, and (2) one based upon *Russell's simplified theory of types*. The reasons for choosing these language-systems are threefold. First they are both extremely important language-systems which have attracted wide attention and have been before the learned world for some years. They both contain vast portions of mathematics, and neither is known to be inconsistent. And secondly, a meta-language such as

[1] E. V. Huntington, "A Set of Postulates for Real Algebra, comprising Postulates for a One-Dimensional Continuum and for the Theory of Groups", *Transactions of the American Mathematical Society*, 6 (1905), pp. 17–41; and A. Tarski, *Introduction to Logic* (Oxford University Press, New York: 1941), p. 214 and pp. 217–18.

SM_D^L can be taken as the semantical meta-language for either of these kinds of systems, in the same uniform way as for simpler systems. These two kinds of language-systems have occupied a somewhat anomalous position in the more usual ways of formulating semantics.[1] The methods used for simpler object-languages are not applicable without extension, and hence some difficulties must be overcome before we have a semantics at all for languages of these kinds.

A third reason for considering these powerful languages in detail rather than narrower languages is that most well-developed scientific or philosophical languages can be regarded as one or the other of these languages appropriately extended by the addition of further non-logical primitive constants. Most well-developed scientific or philosophical languages contain a good deal of mathematics. Hence it seems natural to consider here very powerful systems providing for practically all the mathematics needed. We could of course consider weaker systems which for specific purposes might require extension. But here no complications will arise from considering the more powerful language-systems *ab initio*.[2]

A word of caution is perhaps needed concerning the phrase 'non-logical'. Some would regard the whole of the Zermelo set theory or of Russell's type theory as a logic. On such a view the axioms of these languages would then also be called *logical* axioms. (See § G below.) But these are then logics in a much broader sense of the term. Throughout this book, we construe 'logic' in the very narrow sense discussed in Chapter II. By 'non-logical' is meant throughout 'non-(logical-in-the-elementary-classical-sense)'. In part the decision concerning the extent of the word 'logic' is a matter of convention. The important point is simply to note the very narrow sense in which the word is being used throughout.[3]

In § A a form of *axiomatic set theory* will be sketched somewhat informally. And then, in § B, *formal definitions* of being a set-theoretical axiom and of allied syntactical and semantical concepts will be

[1] See, e.g., A. Tarski, *Der Wahrheitsbegriff*, pp. 363–9, and "On Undecidable Statements in Enlarged Systems of Logic and the Concept of Truth", *The Journal of Symbolic Logic*, 4 (1939), pp. 105–12; J. Kemeny, "Type Theory vs. Set Theory", abstract in *The Journal of Symbolic Logic*, 15 (1950), p. 78; and H. Wang, "Truth Definitions and Consistency Proofs", *Transactions of the American Mathematical Society*, 73 (1952), pp. 243–75. Also see Chapter VII, § E below.

[2] The material of this chapter closely follows the author's "On Truth and Multiple Denotation", *The Journal of Symbolic Logic*, 18 (1953), pp. 11–18. See also "What is a Rule of Language?", *Science, Language, and Human Rights*, Papers of the American Philosophical Association, Eastern Division (University of Pennsylvania Press, Philadelphia: 1952), pp. 105–25.

[3] Cf. especially R. Carnap, *The Logical Syntax of Language, passim.*

given within a formalized meta-language. In § C, an informal sketch of a system based upon the *simplified theory of types* will be given, preparatory to a *more rigorous formulation* in § D. And in § E, the basic semantical properties of that system will be sketched. In § F some remarks concerning the procedure of giving an *alternative interpretation* to a logistic system are put forward. Finally, the so-called *language-systems of higher order* are described very briefly in § G.

The reader who is not interested in these powerful language-systems, or in the alternative semantical theories to be discussed in the next chapter, may go on directly to Chapter VIII, on non-translational semantics. The present chapter and the next are slightly more difficult than those which follow them, and perhaps of less philosophical interest.

A. Zermelo-Skolem System S. The set theory we choose is essentially Church's form of the Zermelo–Fraenkel–Skolem theory.[1] There are also several other kinds of set theory to consider.[2] Each kind is intended to provide implicitly an analysis of the notion of *set*, one of the fundamental notions of modern mathematics. Different kinds of set theory characterize this notion in different ways, and perhaps none of these ways is yet altogether satisfactory. In choosing the Zermelo–Skolem–Church theory, we do not therewith claim it to be the most satisfactory. The theory will not be presented here in great detail, because its study belongs properly to mathematics. We shall be concerned only with its foundations, which, however, will be sketched with some care.[3]

The formal system will be called S. S is merely a special case of L, and the basic logic needed is therefore merely that of L with identity.

[1] See E. Zermelo, "Untersuchungen über die Grundlagen der Mengenlehre I", *Mathematische Annalen*, 65 (1908), pp. 261–81, and Th. Skolem, "Einige Bemerkungen zur Axiomatischen Begründung der Mengenlehre", *Wissenschaftliche Vorträge gehalten auf dem Fünften Kongress der Skandinavischen Mathematiker in Helsingfors vom 4. bis 7. Juli 1922* (Helsingfors: 1923), pp. 217–32.

[2] E.g., that of von Neumann or Bernays or Gödel. See J. von Neumann, "Eine Axiomatisierung der Mengenlehre", *Journal für die Reine und die Angewandte Mathematik*, 154 (1925), pp. 219–40 (and *Berichtigung, ibid.*, 55 (1926), p. 128), and "Die Axiomatisierung der Mengenlehre", *Mathematische Zeitschrift*, 27 (1928), pp. 669–752; P. Bernays, "A System of Axiomatic Set Theory", *The Journal of Symbolic Logic*, 2 (1937), pp. 65–77; 6 (1941), pp. 1–17; 7 (1942), pp. 65–89 and 133–45; 8 (1943), pp. 89–106; 13 (1948), pp. 65–79, and 19 (1954), pp. 81–96; and K. Gödel, *The Consistency of the Continuum Hypothesis* (Princeton University Press, Princeton: 1940). Cf. also the system of Quine's *Mathematical Logic*, revised edition.

[3] For a useful exposition of the subject see A. Fraenkel, *Abstract Set Theory* (North-Holland Publishing Co., Amsterdam: 1953).

The only non-logical primitive predicate constant is symbolized by 'ε'. The atomic formulae are of the form

(1) '$x \varepsilon y$',

where x and y are variables. This could also be written

'εxy',

to conform with the notation of Chapter II, 'ε' being a two-place predicate constant. But (1) is more usual. S contains no primitive individual or functional constants. (For convenience, let us continue to use 'x', 'y', 'z', 'w', and 'u' as the object-language variables rather than just 'x', 'x''', 'x'''', etc.) The variables are to be thought of as ranging over a fundamental domain of *sets*. 'ε' symbolizes a relation called *membership*. A formula such as '$x \varepsilon y$' is to be read 'the set x *is a member of* the set y'.

The non-logical axioms of S provide one way of characterizing a relation ε of set-membership. The leading idea which these axioms incorporate is, very roughly, as follows. We first assume the existence of at least two sets, a *null* set (*S2* below) and an *infinite* set (*S7*). We then assume that, given such and such a set or sets, there is another set (having a given property) which in some way or other depends upon the given set or sets. E.g., *S3* below will stipulate that given any sets x and y there exists a set whose only members are x and y. Given the null set and an infinite set, there exists, according to *S3*, a set whose only members are those sets. In a somewhat similar way, the axioms stipulated by *S4*, *S5*, *S6*, *S8*, *S9*, and *S10* assert the existence of certain sets on the supposition that other sets are already available. In this way, starting with the sets stipulated by *S2* and *S7*, we generate further sets, sets of sets, etc., as needed.

The non-logical axioms of S are given by the following rules. The first stipulates the *Axiom of Extensionality*, which states that given two sets x and y where every member of x is a member of y and conversely, then x and y are identical.

S1. $\vdash (x)(y)((z)(z \varepsilon x \,.\equiv.\, z \varepsilon y) \,.\supset.\, x = y)$.

The next rule stipulates the *Axiom of the Existence of the Null Set*. This axiom states that there is at least one set which has no member.

S2. $\vdash (\mathrm{E}x)(y) \sim y \varepsilon x$.

The third rule stipulates the *Axiom of Pairing*. According to it, given any two sets x and y, there is a set z the members of which are either x or y.

S3. $\vdash (x)(y)(\mathrm{E}z)(w)(w \varepsilon z :\equiv: w = x \,.\mathrm{v}.\, w = y)$.

(Note that each of these rules stipulates that a specific sentence

of S is an axiom. Each of the logical rules presupposed, stated in Chapter II, stipulates an infinity of formulae to be axioms. And some of the non-logical rules below will stipulate an infinity of axioms also. One could state *S2*, e.g., more generally so as to stipulate an infinity of axioms. *S2* could read as follows.

$$\vdash (\mathrm{E}\boldsymbol{x})(\boldsymbol{y}) \sim \boldsymbol{y} \,\varepsilon\, \boldsymbol{x},$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are distinct variables of S. For each choice of the variables here we would have a different axiom. But there is no need for this, because every sentence of the form

$$\text{`}(\mathrm{E}\boldsymbol{x})(\boldsymbol{y}) \sim \boldsymbol{y} \,\varepsilon\, \boldsymbol{x}\text{'},$$

for distinct $\boldsymbol{x}$ and $\boldsymbol{y}$, follows from *S2* by the rules of quantification. Hence this rule is stated in terms of specifically chosen variables.)

The fourth axiom required is the *Axiom of Sums*, given by the following rule.

S4. $\vdash (x)(\mathrm{E}y)(z)(z \,\varepsilon\, y \,.\!\equiv\!.\, (\mathrm{E}w)(z \,\varepsilon\, w \,.\, w \,\varepsilon\, x))$.

Given any set x, there exists at least one set y, whose members are just the members of members of x. The set y here is the *sum* of the set of sets x.

The *Power Axiom* states that given any set x there exists a set y whose members are just the *subsets* of x. One can define

'$\boldsymbol{x} \subset \boldsymbol{y}$' as '$(\boldsymbol{z})(\boldsymbol{z} \,\varepsilon\, \boldsymbol{x} \,.\!\supset\!.\, \boldsymbol{z} \,\varepsilon\, \boldsymbol{y})$', where $\boldsymbol{x}$ and $\boldsymbol{y}$ are any variables and $\boldsymbol{z}$ is any variable distinct from $\boldsymbol{x}$ and $\boldsymbol{y}$.

Hence one can say that a set x is a subset of a set y if and only if for every set z, if z is a member of x then z is a member of y. The Power Axiom is given by the following rule.

S5. $\vdash (x)(\mathrm{E}y)(z)(z \,\varepsilon\, y \,.\!\equiv\!.\, z \subset x)$.

Note that of these five rules, only *S2* stipulates an out-and-out existential axiom. *S3*, *S4*, and *S5* stipulate axioms of a *conditionally* existential kind. They state that, given such and such a set (or sets), there exists a set depending on it (or them) in the ways described.

Two further specific axioms are required. The first, the *Axiom of Choice*, is also a conditionally existential statement. It asserts that, given any non-null set x of non-null sets no two members of which have a set in common, there exists a set which has one and only one set in common with each member of x.

S6. $\vdash (x)((\mathrm{E}y)y \,\varepsilon\, x \,.\, (y)(y \,\varepsilon\, x \,.\!\supset\!.\, (\mathrm{E}z)z \,\varepsilon\, y) \,.\, \sim(\mathrm{E}y)(\mathrm{E}z)(y \,\varepsilon\, x \,.\, z \,\varepsilon\, x \,.\, \sim y = z \,.\, (\mathrm{E}w)(w \,\varepsilon\, y \,.\, w \,\varepsilon\, z)) :\!\supset\!: (\mathrm{E}y)(z)(z \,\varepsilon\, x \,.\!\supset\!.\, (\mathrm{E}w)(u)((u \,\varepsilon\, z \,.\, u \,\varepsilon\, y) \,.\!\equiv\!.\, u = w)))$.

The *Axiom of Infinity* is an out-and-out existential axiom, asserting

that there is at least one set x having the null set as a member and is such that if any set y is a member of x then any set whose only member is y is also a member of x.

S7. $\vdash (\mathrm{E}x)((y)(\sim(\mathrm{E}z)z \varepsilon y \,.\supset.\, y \varepsilon x) \,.\, (y)(y \varepsilon x \,.\supset.\, (z)((w)(w \varepsilon z \,.\equiv.\, w = y) \,.\supset.\, z \varepsilon x)))$.

In addition to these seven rules, we need three further rules of a somewhat different character. Each such rule will stipulate an infinity of axioms.

The first rule is the *Rule of Subset Formation* (*Aussonderungsaxiom.*)

S8. $\vdash (x)(\mathrm{E}y)(z)(z \varepsilon y \,.\equiv.\, (z \varepsilon x \,.\, \boldsymbol{A}))$, where $\boldsymbol{A}$ is any formula of S not containing 'x' or 'y' as free variables.

Given any such sentential function $\boldsymbol{A}$, the rule stipulates an axiom to the effect that given any set x, there exists a set y the members of which are just the members of x for which $\boldsymbol{A}$ holds.

The *Rule of Replacement* (*Ersetzungsaxiom*), which is required for certain portions of transfinite number theory, states in effect that for any set of sets x and for any definable relation such that for every set y which is a member of x there is one and only one set to which y bears the relation, there exists a set z the members of which are just the objects to which members of x bear the relation.

S9. $\vdash (x)((y)(y \varepsilon x \,.\supset.\, (\mathrm{E}z)(w)(\boldsymbol{A} \,.\equiv.\, w = z)) \,.\supset.\, (\mathrm{E}z)(w)(w \varepsilon z \,.\equiv.\, (\mathrm{E}y)(y \varepsilon x \,.\, \boldsymbol{A})))$, where $\boldsymbol{A}$ is a formula not containing 'x' or 'z' as free variables.

Finally, the *Rule of Foundation or Consolidation* (*Axiom der Fundierung*) restricts the domain of sets in effect to those which can be reached in a finite number of steps. It states roughly that given a (definable) property holding of at least one set, there exists at least one set x having that property but such that no member of x has it. This rule is not strictly required, but it facilitates somewhat the deductive development of the system.

S10. $\vdash (\mathrm{E}x)\boldsymbol{A} \,.\supset.\, (\mathrm{E}x)(\boldsymbol{A} \,.\, \sim(\mathrm{E}y)(y \varepsilon x \,.\, \boldsymbol{B}))$, where $\boldsymbol{A}$ is a formula not containing 'y' as a free variable, and $\boldsymbol{B}$ differs from $\boldsymbol{A}$ only in containing free occurrences of 'y' wherever and only where there are free occurrences of 'x' in $\boldsymbol{A}$.

The logical axioms are *R1–R8* and *Abst*, and the rules of inference are *MP* and *Gen*. If desired, identity can be dispensed with as a primitive. It can be defined as follows.

'$\boldsymbol{x} = \boldsymbol{y}$' abbreviates '$(\boldsymbol{z})(\boldsymbol{z} \varepsilon \boldsymbol{x} \,.\equiv.\, \boldsymbol{z} \varepsilon \boldsymbol{y})$', where $\boldsymbol{x}$ and $\boldsymbol{y}$ are any variables and $\boldsymbol{z}$ is any variable distinct from $\boldsymbol{x}$ and $\boldsymbol{y}$.

Then *R7* is immediately provable. In place of *S1* we should now assume that

$$\vdash (x)(y)(z)(x = y :\supset: x \varepsilon z .\supset. y \varepsilon z).$$

Then *R8* is likewise provable.

Also, *S* can be further simplified, in other respects, if desired.[1]

If we were to develop the consequences of *S1–S10* in detail, we should see that all the basic concepts of modern mathematics are in a sense implicitly contained within *S*. Not only is every such concept definable, but the fundamental theorems concerning them are also provable within *S*. To see this explicitly, we should first construct elementary arithmetic, going on to the theory of rational numbers, then on to real, signed real, and complex numbers, proving the basic theorems in each of these areas. We could then go on to topology, geometry, modern algebra, etc., showing in turn how each of these domains of mathematical theory can be accommodated in effect within *S*.

If we should wish to use *S* for purposes other than to develop mathematics, we should add to *S* non-logical primitive predicates appropriate to a particular scientific or philosophical subject-matter. We should then add to the non-logical rules of *S* further non-logical rules characterizing these new primitives. The resulting theory would be the Zermelo set theory as applied to a particular area of science or philosophy. Very little work has been done to date in applying this theory to such areas.

B. Definition of 'Axiom of S' within SM^S_D. Let us now mirror the content of the preceding section within a formalized meta-language. Taking *L* as *S*, we can specialize SM^L_D to provide a semantics for *S*. In this case we speak of SM^S_D, the specific SM^L_D taking *S* as the object-language.

S contains only one primitive predicate constant 'ε'. Let us use '*ep*' within SM^S_D as the structural-descriptive name of this symbol. '*ep*' is used in place of '*pee*', being more suggestive of 'ε'. Also we now write

$$\text{'}(ex \frown ep \frown ex \frown ac)\text{'},$$

and so on, rather than

$$\text{'}(ep \frown ex \frown ex \frown ac)\text{'}.$$

Bearing this slight change in mind, we presuppose the whole of the

[1] See, e.g., W. Ackermann, "Mengentheoretische Begründung der Logik", *Mathematische Annalen*, 115 (1937–8), pp. 1–22; and H. Wang, "On Zermelo's and von Neumann's Axioms for Set Theory", *Proceedings of the National Academy of Sciences*, 35 (1949), pp. 150–5.

syntactical part of SM_D^S before us, developed with only slight changes as in Chapter III. But we must now supply a definition of *non-logical* (or, as we may also say, *set-theoretical*) *axiom*. This can be given as follows.

Corresponding to each of the rules *S1–S10* given above, we have here a definition.

An expression *a* of SM_D^S is an Axiom of Extensionality of *S* if and only if it has the appropriate form of an axiom stipulated by *S1*.

'ExtAx *a*' abbreviates '*a* = (*ex qu ex∩ac qu* (*ex∩ac∩ac qu* (*ex∩ac∩ac∩ep∩ex tripbar ex∩ac∩ac∩ep∩ex∩ac*) *hrsh ex∩id∩ex∩ac*))'.

The correspondence with *S1* seems obvious enough. (Notice that we use here '(*ex∩ac*)' as a structural description of '*y*', '(*ex∩ac∩ac*)' of '*z*', etc., because the use of '*y*', '*z*', etc., in the preceding section was *sensu stricto* illicit!)

In a similar way one can define the following notions:

'NullSetAx *a*' corresponding to *S2*,
'PairAx *a*' corresponding to *S3*,
'SumAx *a*' corresponding to *S4*,
'PowerAx *a*' corresponding to *S5*,
'ChoiceAx *a*' corresponding to *S6*,

and

InfAx *a*' corresponding to *S7*.

In the case of the rule *S8*, the formalized definition is only slightly more complicated. An additional clause is required in the definiens. We can let

'SubsetFormAx *a*' abbreviate '(E*b*)(Fmla *b* . ∼*ex* FV *b* . ∼(*ex∩ac*) FV *b* . *a* = (*ex qu ex∩ac exisqu ex∩ac∩ac qu* (*ex∩ac∩ac∩ep∩ex∩ac tripbar* (*ex∩ac∩ac∩ep∩ex dot b*))))'.

Corresponding to *S9* we let

'ReplAx *a*' abbreviate '(E*b*)(Fmla *b* . ∼*ex* FV *b* . ∼(*ex∩ac∩ac*) FV *b* . *a* = (*ex qu* (*ex∩ac qu* (*ex∩ac∩ep∩ex hrsh ex∩ac∩ac exisqu ex∩ac∩ac∩ac qu* (*b tripbar ex∩ac∩ac∩ac∩id∩ex∩ac∩ac*)) *hrsh ex∩ac∩ac exisqu ex∩ac∩ac∩ac qu* (*ex∩ac∩ac∩ac∩ep∩ex∩ac∩ac tripbar ex∩ac exisqu* (*ex∩ac∩ep∩ex dot b*)))))'.

And corresponding to *S10*, the Axiom der Fundierung, we let

'FoundAx *a*' abbreviate '(E*b*)(E*c*)(Fmla *b* . *c* $\mathrm{SF}^{(ex\cap ac)}_{ex}$ *b* . ∼(*ex∩ac*) FV *b* . *a* = (*ex exisqu b hrsh ex exisqu* (*b dot tilde ex∩ac exisqu* (*ex∩ac∩ep∩ex dot c*))))'.

An expression of S is a *set axiom* if it is stipulated by one of the rules *S1–S10*.

'SetAx a' abbreviates '(ExtAx a .v. NullSetAx a .v. PairAx a .v. SumAx a .v. PowerAx a .v. ChoiceAx a .v. InfAx a .v. SubsetFormAx a .v. ReplAx a .v. FoundAx a)'.

Strictly, we should write here '$\text{SetAx}_S\ a$', because the notion being defined is relative to the language-system S. But a similar remark would apply equally well to all the predicates relativized to S, either syntactical or semantical, which have been or will be introduced. Thus it is not necessary to introduce the subscript 'S' here, but it must be borne in mind that the notion being defined here is that of being a *set axiom of* S. And similarly throughout.

The definition of being a *theorem* of S may be defined by framed ingredients as in (III,F), and 'Thm a' will express that a is a theorem of S.

With these slight readjustments and additional definitions, the various semantical concepts defined above, including a truth-concept, are now definable in SM_D^S essentially as in Chapters IV and V. A proof of the consistency of S can be given within SM_D^S, as well as a proof of the consistency of SM_D^S relative to that of the joint language consisting of S together with M formulated as a syntax language of S.

C. Theory of Types. By specializing it in a slightly different way, SM_D^L may also be taken as a semantical meta-language for an object-language based upon Russell's simplified theory of types.[1] In this case let us speak of the meta-language as SM_D^T, and the object-language as T. Here some further slight changes must be made in the material of Chapters III–V. In this section, we sketch T roughly, preparatory to a more rigorous formulation below.

The subdivision of the variables (or of the entities over which they range) into *types* or *levels* is the leading idea of type theory. The variables of each type are to be regarded as ranging over an appropriate kind of entity. The variables of the first or lowest type range over some totality of objects. The variables of second type then range over *classes* of these objects. Variables of third type range over *classes of*

[1] B. Russell, "Mathematical Logic as Based on the Theory of Types", *American Journal of Mathematics*, 30 (1908), pp. 222–62; A. N. Whitehead and B. Russell, *Principia Mathematica*; F. P. Ramsey, *The Foundation of Mathematics*, pp. 1–61; and L. Chwistek, "Antynomje logiki formalnej", *Przegląd Filozoficzny*, 24 (1921), pp. 164–71. The formulation of simplified type theory in the present chapter is essentially that of Tarski. See *Der Wahrheitsbegriff*, pp. 364–6.

classes of these objects. And so on. An exact specification of the entities taken as the fundamental objects need not be given here. Once these are specified, the classes of them, classes of classes of them, and so on, are then determined. The variables of any type range only over the entities of that type, and no variables are admitted which range over *all* the entities considered. The treatment of variables is more complicated for T than for S because of this distinction of type. But T may still be regarded as a first-order language with an infinity of different styles of variables ranging over an infinity of different kinds of objects. The individuals of T comprise the fundamental objects chosen as well as classes of them, classes of classes of them, and so on. (See § G below.)

Some logicians think that the distinction of type has a certain "naturalness" and corresponds more or less roughly with intuition. Others find it somewhat *ad hoc* and cumbersome. However this may be, type theory has come to be accepted as one of the most important methods of providing a logical basis for mathematics and natural science.

This theory was first developed as a method of avoiding paradox or contradiction. Without the distinction of type, or some alternative method (such as that of S above), one may in fact run into contradiction. Suppose we naïvely define a given class of classes K as the class of just those *classes which are not members of themselves*. By considering now whether K is itself a member of itself, we run into contradiction. For suppose K is a member of itself. It is then one of the classes which are not members of themselves, and hence is *not* a member of itself! On the other hand, if we suppose K is not a member of itself, K is then *not* one of those classes which are *not* members of themselves, and hence *is* a member of itself! This is, roughly speaking, the famous Russell paradox. On the basis of the theory of types, this paradox is avoided by requiring that no class can significantly be or not be a member of any class of the same type. In any system based upon type theory there is, then, an important restriction upon the formulae regarded as significant or meaningful.

In T, the only non-logical primitive predicate constant is 'ε', and the atomic formulae are of the forms

$$'x \varepsilon y',$$

where y is a variable of type one higher than that of x, or

$$'x = y',$$

where x and y are variables of the same type. These restrictions are incorporated into the definition of 'formula of T' (see § D below), and many kinds of expressions (which one might have supposed to be

formulae) are rejected as meaningless. Note that we regard 'ε' here as a *non-logical* predicate constant, construing 'logic' in the narrow sense described above. (See, however, § G below.) Also note that T contains no primitive individual or functional constants.

In T, let the m-th variable of type n be written

(1) $$\text{`}x\overbrace{'''\ldots'}^{n \text{ times}}\underbrace{''''\ldots'}_{m \text{ times}}\text{'}.$$

The accents or superscripts indicate type level, and the subscripts serve merely to differentiate between variables.[1] For convenience, let expressions such as (1) be written as

(2) $$\text{`}x^{\boldsymbol{n}}_{\boldsymbol{m}}\text{'}.$$

'$\boldsymbol{n}$' here stands for a string of n accents as superscripts and '$\boldsymbol{m}$' for a string of m accents as subscripts. Expressions such as (2) are metalinguistic expressions standing for variables of T. Occasionally we may write '$x^{\boldsymbol{n}}_{\boldsymbol{m}}$' in place of (1), as the object-language variable stipulated by (2). This notational simplification will facilitate many statements below.

The basic logical substructure is that of Chapter II. *R1–R8*, *Abst*, *MP*, and *Gen* simply reappear here as rules of T, but with certain restrictions and changes due to the distinction of type. *R5* must now read as follows.

$\vdash (x^{\boldsymbol{n}}_{\boldsymbol{m}})\, \boldsymbol{A} \,.\supset.\, \boldsymbol{B}$, if $\boldsymbol{B}$ differs from $\boldsymbol{A}$ only in containing free occurrences of some one variable '$x^{\boldsymbol{n}}_{\boldsymbol{k}}$' wherever there are free occurrences of '$x^{\boldsymbol{n}}_{\boldsymbol{m}}$' in $\boldsymbol{A}$.

The exhibited variables in *R6*, *R7*, and *Gen* may be of any type. But in *R8* the two exhibited variables must be of the same logical type. The Axioms of Abstraction cannot be stated here in quite so general a form as above in (II,D). Here we have rather the following rule.

Abst. $\vdash x^{\boldsymbol{n}}_{\boldsymbol{m}} \ni \boldsymbol{B}\, x^{\boldsymbol{n}}_{\boldsymbol{k}} .\equiv.\, \boldsymbol{A}$, if '$x^{\boldsymbol{n}}_{\boldsymbol{m}}$' is a variable distinct from '$x^{\boldsymbol{n}}_{\boldsymbol{k}}$' and not occurring freely in $\boldsymbol{A}$, and $\boldsymbol{B}$ differs from $\boldsymbol{A}$ only in containing free occurrences of '$x^{\boldsymbol{n}}_{\boldsymbol{m}}$' in 0 or more places where there are free occurrences of '$x^{\boldsymbol{n}}_{\boldsymbol{k}}$' in $\boldsymbol{A}$.

Just four *non-logical* rules are needed. The first stipulates the *Axioms of Extensionality*, one for each type level.

T1. $\vdash (x_1^{\boldsymbol{n}+\boldsymbol{1}})(x_2^{\boldsymbol{n}+\boldsymbol{1}})((x_1^{\boldsymbol{n}})(x_1^{\boldsymbol{n}}\, \varepsilon\, x_1^{\boldsymbol{n}+\boldsymbol{1}} .\equiv.\, x_1^{\boldsymbol{n}}\, \varepsilon\, x_2^{\boldsymbol{n}+\boldsymbol{1}}) .\supset.\, x_1^{\boldsymbol{n}+\boldsymbol{1}} = x_2^{\boldsymbol{n}+\boldsymbol{1}})$.

[1] This usage is due essentially to Tarski. In the material of the preceding chapters, superscripts or accents were used to differentiate between variables. That usage should not be confused with their usage here to indicate type level.

These axioms assert that if two classes of the same type have the same membership, they are identical.

The *Axioms of the Existence of Classes* state that there exists a class of appropriate type corresponding to any sentential function of at least one variable. The rule stipulating these axioms is as follows.

T2. $\vdash (\mathrm{E}x_1^{n+1})(x_1^n)(x_1^n \varepsilon x_1^{n+1} .\equiv. A)$, if A is a formula not containing 'x_1^{n+1}' as a free variable.

Also forms of an *Axiom of Infinity* and of *Axioms of Choice* are needed. Both *T1* and *T2* stipulate an infinity of axioms, one axiom for each specific determination of 'n'. We need assume, however, only one axiom of infinity, providing for the existence of at least one infinite class of classes of individuals. The existence of infinite classes of higher types then follows. The Axiom of Infinity asserts that there exists at least one non-null class x_1^3 of classes of individuals, every member of which contains a proper subclass which is also a member of x_1^3. Let

$$\text{'}x_k^{n+1} \subset x_m^{n+1}\text{' abbreviate '}(x_q^n)(x_q^n \varepsilon x_k^{n+1} .\supset. x_q^n \varepsilon x_m^{n+1})\text{'}.$$

To say that a class x_k^{n+1} is a *subclass* of a class x_m^{n+1}, is to say that every member of x_k^{n+1} is a member of x_m^{n+1}. The Axiom of Infinity is then stipulated by the following rule.

T3. $\vdash (\mathrm{E}x_1^3)((\mathrm{E}x_1^2)x_1^2 \varepsilon x_1^3 . (x_1^2)(x_1^2 \varepsilon x_1^3 .\supset. (\mathrm{E}x_2^2)(x_2^2 \varepsilon x_1^3 . x_2^2 \subset x_1^2 . \sim x_1^2 \subset x_2^2)))$.

Finally, axioms of choice are assumed in the form of Russell's *Multiplicative Axioms*. These tell us that given any non-null class x_1^{n+2} of mutually exclusive classes (of type $n+1$) there exists at least one class which contains one and only member in common with each non-null member of x_1^{n+2}.

T4. $\vdash (x_1^{n+2})((\mathrm{E}x_1^{n+1})x_1^{n+1} \varepsilon x_1^{n+2} . \sim(\mathrm{E}x_1^{n+1})(\mathrm{E}x_2^{n+1})(x_1^{n+1} \varepsilon x_1^{n+2} . x_2^{n+1} \varepsilon x_1^{n+2} . \sim x_1^{n+1} = x_2^{n+1} . (\mathrm{E}x_1^n)(x_1^n \varepsilon x_1^{n+1} . x_1^n \varepsilon x_2^{n+1})) :\supset: (\mathrm{E}x_1^{n+1})(x_2^{n+1})(x_2^{n+1} \varepsilon x_1^{n+2} . (\mathrm{E}x_2^n)x_2^n \varepsilon x_2^{n+1} :\supset: (\mathrm{E}x_1^n)(x_2^n)(x_2^n \varepsilon x_2^{n+1} . x_2^n \varepsilon x_1^{n+1} :\equiv: x_2^n = x_1^n)))$.

Here also, as in S, identity can be dropped as a primitive, being definable as follows.

$$\text{'}x_m^n = x_k^n\text{' abbreviates '}(x_p^{n+1})(x_m^n \varepsilon x_p^{n+1} .\supset. x_k^n \varepsilon x_p^{n+1})\text{'}.$$

And here likewise *R7* and *R8* can then be dropped, being provable using the above definition. But *T1* is retained.

Note that T is formulated in such a way as to contain no variables for relations or functions, but only for classes, classes of classes, etc. Relations and hence functions can, nevertheless, be handled within

T, using the devices of Wiener and Kuratowski.[1] The leading idea of these devices is to regard relations as classes of *ordered pairs* of a certain kind. Suppose we wish to gain the effect within T of a variable 'R' for dyadic relations between individuals, and to express that some individual x^1_m bears R to x^1_o. We form first the ordered pair of x^1_m and x^1_o as a class of two classes, one of them being the class whose only member is x^1_m, the other being the class whose only members are x^1_m and x^1_o. Then x^3_n is such an ordered pair if

(3) $(x^2_1)(x^2_1 \varepsilon x^3_n :\equiv: (x^1_3)(x^1_3 \varepsilon x^2_1 .\equiv. x^1_3 = x^1_m) .\text{v}. (x^1_3)(x^1_3 \varepsilon x^2_1 :\equiv: x^1_3 = x^1_m .\text{v}. x^1_3 = x^1_o)).$

The effect of 'R' as a variable can be achieved by using a variable 'x^4_n' of fourth type. To say that x^1_1 bears R to x^1_2 is to say that

(4) $(x^3_1)((x^2_1)(x^2_1 \varepsilon x^3_1 :\equiv: (x^1_3)(x^1_3 \varepsilon x^2_1 .\equiv. x^1_3 = x^1_1) .\text{v}. (x^1_3)(x^1_3 \varepsilon x^2_1 :\equiv: x^1_3 = x^1_1 .\text{v}. x^1_3 = x^1_2)) .\supset. x^3_1 \varepsilon x^4_n).$

This may conveniently be abbreviated as

$$\text{`}\langle x^1_1, x^1_2\rangle \varepsilon x^4_n\text{'}.$$

The treatment of relations of higher degree is as follows. The *ordered triple* of individuals x^1_1, x^1_2, and x^1_3, e.g., may be identified with the ordered pair of pairs, the first of which is the ordered pair of x^1_1 with x^1_2, and the second, of x^1_2 with x^1_3. The effect of a variable 'R' for a triadic relation is then achieved by a variable of *sixth* type. Let (3) be abbreviated by 'x^3_n OP x^1_m, x^1_o'. To say that R holds among x^1_1, x^1_2, and x^1_3 is then to say that

$(x^5_1)((x^4_1)(x^4_1 \varepsilon x^5_1 :\equiv: (x^3_1)(x^3_1 \varepsilon x^4_1 :\equiv: x^3_1 \text{ OP } x^1_1, x^1_2) .\text{v}. (x^3_1 \varepsilon x^4_1 :\equiv: x^3_1 \text{ OP } x^1_1, x^1_2 .\text{v}. x^3_1 \text{ OP } x^1_2, x^1_3)) .\supset. x^5_1 \varepsilon x^6_n).$

This may be abbreviated as

$$\text{`}\langle x^1_1, x^1_2, x^1_3\rangle \varepsilon x^6_n\text{'}.$$

And so on for relations of still higher degree. The treatment of relations between or among objects of higher type is similar.

All the ordered pairs considered have been *homogeneous* in the sense that the members of the pair are of the same logical type. But *heterogeneous* ordered couples may easily be introduced. E.g., suppose we wish to form the ordered pair of x^2_1 with x^1_1. This we may regard as the ordered pair of x^2_1 with the class whose only member is x^1_1, and

$$\text{`}\langle x^2_1, x^1_1\rangle \varepsilon x^5_n\text{'}$$

may abbreviate

[1] See N. Wiener, "A Simplification in the Logic of Relations", *Proceedings of the Cambridge Philosophical Society*, 17 (1912–14), pp. 387–90, and C. Kuratowski, "Sur la Notion de l'Ordre dans la Théorie des Ensembles", *Fundamenta Mathematicae*, 2 (1921), pp. 161–71.

$(x_1^4)((x_1^3)(x_1^3 \varepsilon x_1^4 :\equiv: (x_2^2)(x_2^2 \varepsilon x_1^3 .\equiv. x_2^2 = x_1^2) .\text{v}. (x_2^2)(x_2^2 \varepsilon x_1^3 :\equiv: x_2^2 = x_1^2 .\text{v}. (x_2^1)(x_2^1 \varepsilon x_2^2 .\equiv. x_2^1 = x_1^1))) .\supset. x_1^4 \varepsilon x_n^5)$'.

In a similar way we can define expressions such as

$$\text{`}\langle x_1^3, x_1^1\rangle \varepsilon x_n^6\text{'},$$
$$\text{`}\langle x_1^2, x_1^1, x_2^1\rangle \varepsilon x_n^7\text{'},$$

and so on, accommodating heterogeneous couples, triples, etc., of greater complexity. (There are to be sure other methods of handling ordered couples, triples, and so on, which we need not consider here.)

For the deductive development of what is essentially the system T, the reader is referred to *Principia Mathematica*. As in the case of S, most of modern mathematical theory can be accommodated in T. But the two systems T and S are not equivalent, for it is known that S is in certain respects more powerful than T.[1]

Here, also, as in the case of S, we can add new non-logical primitives and axioms characterizing them, thereby formalizing given areas of scientific or philosophical theory.

D. 'Axiom of T' within SM_D^T. The foregoing rough sketch of T will now be made more precise. Taking L as T, we specialize SM_D^L to provide a semantics for T, the resulting meta-language being SM_D^T.

Because of the treatment of variables in T we need an additional structural-descriptive primitive '*subscr*' in SM_D^T for the primitive subscript. '(*ex*⌢*ac*⌢*subscr*)', e.g., becomes the structural description of the variable 'x','. Let '*ep*' also be the structural description here of 'ε', as in SM_D^S. Hence within SM_D^T

'PS a' now abbreviates '(a = *lp* .v. a = *rp* .v. a = *vee* .v. a. = *tilde* .v. a = *ex* .v. a = *ac* .v. a = *subscr*. v. a = *id* .v. a = *invep* .v. a = *ep*)'.

In the syntactical part of SM_D^T the definition of 'Vbl a' must differ slightly from that given in (III,D). Here a variable is *ex* followed by a string of superscripts or accents followed by a string of subscripts. The notion of being a *string of subscripts* is akin to that of being a string of accents, and thus

'SubscrString a' may abbreviate '(b)(PS b . b Seg a :⊃: b = *subscr*)'.

We can then define the notion of being a variable, by letting

'Vbl a' abbreviate '(Eb)(Ec)(AcString b . SubscrString c . a = (*ex*⌢ b⌢c))'.

(The notion of being a variable of such and such a type level will be defined in a moment.)

[1] See J. Kemeny, *op. cit.*

The definition of being an *atomic formula* of T also must be given in a slightly different way from that of (III,D). We can say that an expression a is an atomic formula of T if and only if (i) it consists of ex concatenated with a string of accents b concatenated with a string of subscripts c concatenated with ep, then with ex, then with b, *then with* ac, and finally with any string of subscripts d, or (ii) a is of the identity form, ex followed by b, then by c, then by id, then by ex, then by b, and then by d.

'AtFmla a' abbreviates '(Eb)(Ec)(Ed)(AcString b . SubscrString c . SubscrString d : $a = (ex \frown b \frown c \frown ep \frown ex \frown b \frown ac \frown d)$ .v. $a = (ex \frown b \frown c \frown id \frown ex \frown b \frown d)$)'.

Note especially the condition of significance here for atomic formulae a containing ep. a must consist of a variable followed by ep followed by a variable of next higher type.

The definition of 'Fmla' can now be given essentially as in (III,D) by means of framed ingredients. But 'Abst', which this definition presupposes, must be defined a little differently because the two variables involved must be of the same type. The definiens of the definition of 'AtFmla' just given suggests that we may define the notion that two variables a and b are of the *same type* as follows.

'a SmTp b' abbreviates '(Ee)(Ec)(Ed)(AcString e . SubscrString c . SubscrString d . $a = (ex \frown e \frown c)$. $b = (ex \frown e \frown d)$)'.

We can then define

'a Abst b' as '(Ec)(Ed)(c SmTp d . $a = (c \frown invep \frown b \frown d)$)'.

The definition of 'Fmla' can then be given essentially as in (III,D).

Some further slight changes must be made in some of the other definitions. In the definiens of the definition of 'SF1' in (III,E), the clause 'b SmTp c' must be added as an additional conjunct. (The subsequent definitions using 'SF1' adjust themselves automatically.) The definition of 'Axiom of T' can then be given rigorously within SM_D^T. Any further slight changes required may easily be given.

E. Semantical Notions within SM_D^T. Because of the distinction of type the semantical material of Chapters IV–V must be slightly altered.

The rules of SM_D^T are essentially as above in (IV,D), although some slight changes are required in the statement of the Rules of Denotation. Here the first rule, *DenR1*, reads as follows.

$\vdash (x_m^n)(a \text{ Den } x_m^n \mathrel{.\equiv.} \text{---}x_m^n\text{---})$, where '---$x_m^n$---' is any sentential function of the one variable 'x_m^n' from the translational part

of SM_D^T, and in place of 'a' we put in the structural description of the abstract '$x_m^n \ni \text{---} x_m^n \text{---}$'.

Also the Restrictive Rule, *DenR2*, must be changed slightly. Recall that this rule states that

$$\vdash a \text{ Den } x \,.\supset.\, \text{PredCon } a.$$

In place of 'x' we put here 'x_m^n'. But also we must specify in the conclusion that a be a predicate constant of such and such an order or type. Roughly, an abstract of the form '$x_m^n \ni \text{---} x_m^n \text{---}$' will be said to be of type $n+1$. To define this formally we first define the notion of being a *variable* of type n. The use of numerical variables throughout is merely a linguistic convenience; they replace strings of accents and subscripts. We cannot define in general the notion that an expression a is a variable of type n. But we can define the notions that an expression is a variable of first type, of second type, etc.

'Vbl′ a' or 'Vbl^1 a' abbreviates '(Eb)(SubscrString b . a = ($ex \frown ac \frown b$))',

'Vbl″ a' or 'Vbl^2 a' abbreviates '(Eb)(SubscrString b . a = ($ex \frown ac\ ac \frown b$))',

and so on. Where 'n' stands for any *fixed* string of accents, we can in effect presuppose 'Vbl^n a' as defined. We can then also define the notion of being a predicate constant of type $n+1$ (strictly, of type

$$\underbrace{''''\ldots'}_{n+1 \text{ times}}\ ).$$

'PredCon^{n+1} a' abbreviates '(Eb)(Ec)(Vbl^n b . SentFuncOne c,b . a = ($b \frown invep \frown c$))'.

Bearing these definitions in mind, we can state *DenR2* as follows.

$$\vdash a \text{ Den } x_m^n \,.\supset.\, \text{PredCon}^{n+1}\, a.$$

The truth-concept for T cannot be defined as in Chapter V. The reason is that the notions of being a variable, of being a universal predicate constant, etc., now give way to an infinity of related notions, one for each type. In place of 'Univ b' we now have an infinity of notions, 'Univ″ b', 'Univ‴ b', etc. (We have no 'Univ″.) Also the various translational variables must now have accents or superscripts attached to indicate type. 'Tr_1', as defined by (I) of (V,A), gives way to an infinity of truth-predicates, 'Tr_1''', 'Tr_1'''', etc. 'Tr_1'' a', e.g., may be introduced as an abbreviation for

$$\text{'(AtUnivSent } a \,.\, (\text{E}b)(\text{Univ}''' \, b \,.\, b \, \text{R}_{rp}^{invep}\, a))\text{'}.$$

But (II) of (V,A) presupposes a *single* notion Tr_1 and not an infinity of notions, and hence cannot be given here. For similar reasons (III),

(IV), and (V) of (V,B) cannot be carried over without change. But we can gain adequate truth-definitions for T in SM_D^T in several ways as follows.

First we can reconstrue (III) of (V,B), the definition in terms of prenex normal forms. From (II,G) we recall that given any sentence a, a sentence in prenex normal form equivalent to a may be found containing a universal quantifier as its left-most quantifier. Within T, we can also require that the variable contained within this universal quantifier be of some fixed type, say of type 1. If we reconstrue 'PNF' in this way, (III) of (V,B) gives way to the following definition.

'Tr a' abbreviates '(Sent a . (Eb)(Ec)(b PNF a . Univ'' c . c R_{rp}^{invep} b))'.

This truth-definition can be proved adequate essentially as in (V,C).

In a similar way definitions analogous to (IV) and (V) of (V,B) can be given. Suppose we pick out a type in advance, say the nth. Then, using essentially (IV), we can define

'Tr a' as '(Sent a . (Eb)(Vbl$\overbrace{''''\cdots''}^{n \text{ times}}$ b . Univ$\overbrace{''''\cdots'''}^{n+1 \text{ times}}$ ($b \frown$ *invep* $\frown$ (a *dot* $b \frown$ *id* $\frown b$))))'.

In this way we gain an infinity of different truth-definitions, one for each n, each one of which can be proved adequate. Also analogous definitions can be given on the basis of (V) of (V,B) rather than (IV).

So far as gaining an adequate truth-concept for T is concerned, the definitions above show clearly that a very restricted relation of multiple denotation would suffice. We could in fact regard 'a Den $x_{\boldsymbol{m}}^{\boldsymbol{n}}$' as significant only where $\boldsymbol{n}$ is a *single* superscript. *DenR1* and *DenR2* need then be assumed only for this $\boldsymbol{n}$. The variable 'b' in the definition just given would then be of first type. An adequate truth-concept for T is definable, as we have seen, on this basis and therewith the other basic notions of semantics. However, it may be desirable for other purposes to have 'Den' available in a wider sense.

The consistency of SM_D^T can be proved relative to that of T and M jointly, essentially as in (V,G). (Hence also the semantical metalanguage for T based on the restricted relation of multiple denotation mentioned is relatively consistent.)

Note that within T, and hence within its translation within SM_D^T, the one-place abstracts need no longer be regarded as standing for just *virtual* classes but may if desired stand for genuine classes as values for variables. If abstracts are so regarded, they become *terms* of T and hence legitimate substituends for variables. Certain slight changes must then be made in the preceding material. Note in particular that *T2*, providing the Axioms for the Existence of Classes, is then provable.

F. Re-Interpretation. It has been assumed throughout in the discussion of the various semantical meta-languages that the values of the translational variables are the same as those of the object-language L. Of course, this way of speaking is somewhat misleading, because it suggests that L is first given and the semantical meta-language constructed afterwards. What is intended is that L is first given somewhat informally, and the semantical meta-language is then constructed in such a way as to provide the syntax and semantics of L in a precise form. There is thus no question of the priority of one over the other. Strictly we have no L except as formulated in a precise form. But also we have no semantical meta-language except as relativized to a given L.

Ordinarily there are several alternative interpretations that may be given to the same underlying logistic system. Given one interpretation of L, we shall speak of *re-interpretation* as the procedure of giving an alternative interpretation. Re-interpretation is merely interpretation over again. But the domains of values for variables may differ in two interpretations and the primitive predicate constants may denote objects of different kinds. The new domain may be a sub-class or a super-class of the old domain or it may overlap with it or the two may be disjoint altogether. The objects denoted by the primitive predicate constants are determined accordingly. But note that the logical symbols of L, '$\sim$', 'v', etc., remain fixed. These function in the new interpretation essentially as they did in the old one. Their meaning, so to speak, is not subject to change. Of course we might wish to assign new meanings to these symbols as well. But this would involve a more radical kind of re-interpretation than that under consideration.

Of the various interpretations that can be given to a logistic system, one is usually regarded as the basic or *standard* one. (This has already been suggested in (I,B).) The other interpretations are then in some sense non-standard. Especially important non-standard interpretations for such systems as S and T exist, in which the fundamental domain of objects is the class of natural numbers (or any other denumerable domain). (See (XIII,I).) However, if we were to formulate S or T as interpreted within the domain of natural numbers, we should proceed exactly as above, in SM^S_D or SM^T_D. But the range of the translational variables in these meta-languages would now be construed as the domain of natural numbers.

To make precise the procedure of re-interpretation would require a considerable enrichment in the meta-languages here. In particular we should need an account of the notion of the *range* of variables. Speaking very roughly and informally, we can define the range of the

variables of L as such and such a class or virtual class of objects. This class we may call 'ValVbl_L', the class of values for variables of L. Within the meta-languages above ValVbl_L was identified with the range of the translational variables. But if re-interpretation is to be described precisely within a semantical meta-language this identification would not be desirable. In the re-interpretation, in fact, we should wish to consider also a new domain of values of variables, say ValVbl'_L, defined (again) as such and such a class or virtual class of objects. The semantical meta-language here would contain expressions for objects of various kinds as well as the means of distinguishing them from each other.

We need not discuss this kind of a meta-language any further, because it is essentially merely an extended version of SM^L_D. But for some philosophical purposes such an extended meta-language would be useful. (For additional comments concerning re-interpretation, in non-translational semantics, see (XIII, H–I).)

G. Language-Systems of Finite Order. In the preceding sections T was formulated as a first-order language-system. That is to say, 'ε' was taken as a non-logical primitive and the individuals were taken as comprising the fundamental objects (of lowest type) *together with* all classes of them, all classes of classes of them, and so on. T was simply a special case of the L's of Chapter II. The basic logic presupposed was the first-order logic of that chapter. T was formulated as a mathematical theory of classes, just as S was formulated as a mathematical theory of sets.

Often systems such as T are formulated, not as mathematical theories based upon first-order logic, but as a higher kind of logic containing first-order logic as a part. Let us consider this alternative way of formulating T a little more closely. We shall not give an exact formulation, but only a few rough, explanatory remarks.

'Logic', as we have noted again and again, is used in this book only in the most elementary (classical) sense. This is perhaps the sense of the word as most commonly used by mathematicians. Some logicians prefer a broader meaning of the word in order to include under it some or perhaps all of the theory of classes construed on the basis of the theory of types. In this broader sense, the 'ε' of type theory becomes a *logical* primitive and the theory governing it a logical theory. If the theory is regarded as a logic, it is desirable not to use 'ε', but to write

$$\text{‘}x_j^{n+1}(x_k^n)\text{’ or ‘}x_j^{n+1}x_k^n\text{’}$$

in place of

$$\text{‘}x_k^n \,\varepsilon\, x_j^{n+1}\text{’}.$$

In the informal explanation of first-order languages, in Chapter II, the word 'individual' was antecedently understood. The word was used there roughly in the sense of 'entity from some non-null, clearly demarcated totality of objects'. Let us speak for the moment of individuals in this sense as *individuals*$_1$. In the formulation of T above the individuals$_1$ of T are taken as the entities of lowest type together with all classes of them, all classes of these classes, and so on. These classes, classes of classes, etc., are merely a special kind of individual$_1$. Let us speak of the entities of T of *lowest type* as *individuals*$_2$. Here, in the informal formulation of T as a logic, in addition to understanding 'individual$_2$' in this sense, we must also know what it means to speak of *classes* of such individuals. These classes are not to be construed as a special kind of individual$_1$, as in the formulation of T above, but as an entity of an altogether new kind. And likewise for the classes of classes, and so on. In the logical formulation of T, 'class' is given antecedent meaning, so to speak, in the informal metalanguage, whereas in the mathematical formulation it is not. In the mathematical formulation the axioms characterize classes as a special kind of individual$_1$, and we mean by 'class' then whatever the axioms compel us to mean. The word 'class', so to speak, here has meaning only relative to the axioms. But in the formulation of T as a logic we are already to understand 'class', 'class of classes', etc., as standing for special kinds of objects. The meanings here are not relative to the axioms but rather are fixed in advance.

Hereafter we shall refer to the formulation of T as a logic as T_ω.

It is clear from the foregoing that in essential respects T and T_ω are alike. The difference, however, is not merely terminological. T_ω is a system explicitly interpreted as a logic of classes. It is therefore not subject to re-interpretation. We are not free to reconstrue the relation of membership in some other way, because its meaning is already fixed as a basic logical relation. We are no more free to re-interpret the class-membership connective than we are to re-interpret 'v' or '$\sim$'. The 'ε' of T, however, we are free to re-interpret in any way we wish, conformable with the axioms, as has already been suggested.

In both T and T_ω we have presupposed that the theory of relations is developed as a special branch of the theory of classes, using the devices of Wiener and Kuratowski. This made possible a very simple formulation of T. Often it is useful to develop T or T_ω in a slightly more complex way so as to admit also relations as values for variables together with quantifiers upon them. Not all kinds of relational variables need be admitted. If desired, in addition to the class variables of all types, we may admit, e.g., variables ranging over just

dyadic relations between individuals together with the quantifiers upon them. Of course, the devices of Wiener and Kuratowski show that the effect of such relational variables and quantifiers can be gained in other ways within T or T_ω. Nonetheless, if T or T_ω are to be used as a basis for a scientific or philosophical theory in which certain dyadic relations between individuals are of especial importance, it is frequently convenient to have such relations as values for variables. We shall, then, admit also such variables (and quantifiers) if circumstances require them. (See (VII,E).) And on similar grounds, still other kinds of relational variables may be admitted.

For some purposes it is useful to employ not the whole of T_ω, but only a part of it. We might admit not all the different kinds of class or relational variables but only some variables of consecutive lower types starting with the lowest. We should then have several different theories to distinguish, depending specifically upon what types of variables and quantifiers are admitted.

Let us consider first a system containing only variables of lowest type. We may suppose that we have identity of individuals as primitive, but we should have no atomic formulae such as

$$`x_1^2 x_1^1',$$

these presupposing variables of two consecutive types. But non-logical primitive predicates can be admitted and we then gain a language-system of first order of the kind discussed in Chapter II, but with no primitive individual or functional constants.

Suppose we admit variables of the second type in addition to those of the first. We gain then a system of *second order*.[1] Strictly, there are several systems of second order to be distinguished depending upon the kinds of relational variables, if any, admitted. In the next chapter (§ E) we shall formulate a semantical meta-language of second order. In this semantical meta-language we shall allow, in addition to the class variables, only variables over dyadic relations, and non-logical primitives (other than 'Den') similar to those of SM_D^L. Language-systems of second order, containing also relational variables, have been shown to be very useful languages for mathematics.[2] The underlying logic of such languages is a great deal more powerful than that of first-order languages. Here we can quantify over classes and relations, and in terms of such quantifiers many notions are definable that cannot be defined if such quantifiers are not available. Appropriate Rules of Quantification covering all the kinds of variables

[1] See A. Church, *op. cit.*, pp. 99–104. Church's classification, however, differs in important respects from that used here.

[2] See Hilbert and Bernays, *op. cit.*, Supp. IV, vol. 2, pp. 451–95.

admitted must be added to the logical rules. The non-logical primitive predicates may take as arguments any of the entities over which variables are admitted. There is thus tremendous latitude here in the choice of primitives.

Language-systems of *third* order admit, in addition to variables (and quantifiers) of first and second order, also variables of third order. If among the variables of second order are included variables over dyadic relations, then among the variables of third order one may include, in addition to the variables ranging over classes of classes of individuals, variables ranging over classes of dyadic relations between individuals. Appropriate Rules of Quantification are assumed concerning all the new kinds of quantifiers admitted. And of course, suitable non-logical primitive predicates may be added. These may take as arguments any of the kinds of entities over which variables are admitted.

In a similar way, one goes on to language-systems of *fourth*, *fifth*, etc., order. At each level one admits new variables (and quantifiers) of the proper type, possibly only class variables or possibly relational variables of specified degree also.

These *language-systems of finite order* may all be reconstrued or reformulated as language-systems of first order. This is evident from the remarks above concerning the interrelationship between individuals$_1$ and individuals$_2$. Thus we are not excluding from consideration here systems which in another formulation are of higher order.

Many of the foregoing remarks are vague and imprecise. But this rough account of the language-systems of higher order is all that is required for present purposes. Our only concern hereafter with such systems will be in the next chapter, in connection with an important kind of second-order semantical meta-language.

CHAPTER VII

SATISFACTION, DESIGNATION, DETERMINATION

In this chapter, we drop multiple denotation as a primitive semantical relation and consider three alternatives. One has already been mentioned in (IV,C), the dyadic relation of *satisfaction*. Two further alternatives will be considered, dyadic relations of *determination* and *designation*. The meta-languages here are closely patterned after SM^L_D, the differences being mainly differences of detail. Many of these meta-languages will be shown equivalent with each other. Nonetheless, some interest is thought to attach to these alternative formulations. We shall not develop them in detail but will merely sketch their salient features.

The *semantical meta-language based on satisfaction* will be discussed in § A. The semantical meta-language based upon *designation* will occupy § B, where the object-language is taken as T. And in § C, also taking T as object-language, a semantical meta-language based on *determination* will be sketched. In § D, some *remarks* will be offered as to why semantical meta-languages based on either designation or determination are not appropriate if the object-language is taken as S. In § E we consider the circumstances under which the semantical relations taken heretofore as primitive become *definable*. Finally, in § F some general comments *in favor of* the semantical methods developed thus far in this book are put forward.

A. Satisfaction. We construe satisfaction (symbolized by 'Sat') in the very narrow sense in which it was defined in (IV,C) It was there defined in terms of multiple denotation; but also, it will be recalled, multiple denotation was defined in terms of it. 'x Sat a' is to be read 'the object x satisfies the expression a'. The objects which constitute the domain of this relation are just the objects of the fundamental domain of L. And the expressions which constitute its converse domain are the expressions of L of a certain kind. Hence a semantical meta-language based on 'Sat' need differ only very slightly from that based on 'Den'.

As semantical rules, we now have some *Rules of Satisfaction* in place of the Rules of Denotation (IV,D). The first is closely akin to *DenR1*. It states that for every x, x satisfies an expression a if and

only if ---x---, where 'a' is taken as the structural description of '---x---', a sentential function of (at most) the one variable 'x' from the translational part of the meta-language.

SatR1. $\vdash (\boldsymbol{x})(\boldsymbol{x}\ \text{Sat}\ a\ .\equiv.\ \text{---}\boldsymbol{x}\text{---})$, if (etc., as required).

Recall that the second Rule of Denotation is a restrictive rule, asserting that if an expression denotes anything at all, it is a predicate constant. So also, we need a rule here asserting that if an expression is satisfied by anything at all, it is a sentential function of at most one variable.

SatR2. $\vdash x\ \text{Sat}\ a\ .\supset.\ (\text{E}b)\text{SentFuncOne}\ a,b$.

'Den' being definable in terms of 'Sat', the various semantical notions of SM_D^L can now also be defined in terms of 'Sat'. Rather than simply carry over the definitions of SM_D^L, we should now define directly many of the notions, using 'Sat', *mutatis mutandis*, in place of 'Den'. And among these notions is the truth-concept, defined in an especially simple way as follows.

'Tr a' abbreviates '(Sent a . $(x)x$ Sat a)'.

This truth-definition may be proved adequate by essentially the methods above.

If L is taken as T, a semantical meta-language for T based on 'Sat' may easily be formulated by taking into account the changes required in the treatment of variables. *SatR1* here would be similar to *DenR1* of SM_D^T, and *SatR2* would read as follows.

$$\vdash x_m^n\ \text{Sat}\ a\ .\supset.\ (\text{E}b)(\text{Vbl}^n\ b\ .\ \text{SentFuncOne}\ a,b).$$

From *SatR1* and *SatR2* it is not difficult to prove *DenR1* and *DenR2*. And conversely, from *DenR1* and *DenR2* one can prove within SM_D^L *SatR1* and *SatR2*. Hence in an appropriate sense these two meta-languages, for any given L, are equivalent.

Both these types of meta-languages are applicable to any first-order L in either the narrow or wider sense. We now turn to two methods which are somewhat more limited and which seem appropriate primarily as applied to systems such as T. The first method is based upon designation, the second on determination.

B. Designation in T. Designation is to be construed here in the sense of singular denotation.[1] It is therefore a relation for which the

[1] Semantical meta-languages based on designation have been studied especially by Carnap. Cf. his *Introduction to Semantics*, pp. 49–54 and *passim*; and *Meaning and Necessity*, *passim*. The semantics of this present section was first formulated in the author's "On Types, Denotation, and Truth", *Methodos*, 12 (1951), pp. 308–16. The author is indebted to Professor Alonzo Church for pointing out an error in this paper in his review of it (*The Journal of Symbolic Logic*, 19 (1954), pp. 139–40).

Principle of Univocality holds. This Principle, which states that a denoting expression denotes one and only one object, it will be recalled, was abandoned in the semantical meta-languages above. For T we have strictly not one Principle of Univocality but an infinity of them, one for each type level greater than or equal to two, as we shall see in a moment.

The syntax of T we presuppose developed in the manner of Chapter III, with the requisite adjustments indicated in (VI,D). To construct the semantics of T, we need a primitive symbol for the relation of designation. Let this be 'Des'. We may then say that an expression of T (of a certain kind) designates a *class* (of proper type). Because T contains no individual constants as primitives, we need not be able to say that an expression designates an individual (entity of lowest type). Expressions such as

$$\text{'}a \text{ Des } x^{\boldsymbol{n}}_{\boldsymbol{m}}\text{'}$$

are thus to be meaningful where $\boldsymbol{n}$ is any superscript string of length at least two, but such forms as

$$\text{'}x^{\boldsymbol{n}}_{\boldsymbol{m}} \text{ Des } x^{\boldsymbol{k}}_{\boldsymbol{q}}\text{'}, \text{ or '}x^{\boldsymbol{n}}_{\boldsymbol{m}} \text{ Des } a\text{', or '}a \text{ Des } b\text{'}$$

are regarded as meaningless.

In SM^{L}_{D}, where L is as in Chapter II, there is some one expression which denotes every object of L. In SM^{T}_{D} we have the restrictions of type, and hence only that

$$\vdash (\mathrm{E}a)(x^{\boldsymbol{n}}_{\boldsymbol{m}})a \text{ Den } x^{\boldsymbol{n}}_{\boldsymbol{m}}.$$

Also, in the semantical meta-language based on satisfaction,

$$\vdash (\mathrm{E}a)(x)x \text{ Sat } a.$$

From these three principles respectively, by $TC33$(II), we have also that, in SM^{L}_{D}

$$\vdash (x)(\mathrm{E}a)a \text{ Den } x,$$

within SM^{T}_{D}

$$\vdash (x^{\boldsymbol{n}}_{\boldsymbol{m}})(\mathrm{E}a)a \text{ Den } x^{\boldsymbol{n}}_{\boldsymbol{m}},$$

and within the meta-language based on satisfaction

$$\vdash (x)(\mathrm{E}a)x \text{ Sat } a.$$

But when we come to a semantics for T based on designation we have no analogous principles at all. Not every class of a given type is designated by some expression or other, nor will there be some one expression which designates all classes of a given type level.

In place of the Rules of Denotation or of Satisfaction, we have here some *Rules of Designation*. The first corresponds to *DenR1* or *SatR1*. It states that appropriate expressions designate certain classes.

DesR1. $\vdash (\mathrm{E}x^{\boldsymbol{n}+1}_{\boldsymbol{k}})(a \text{ Des } x^{\boldsymbol{n}+1}_{\boldsymbol{k}} \,.\, (x^{\boldsymbol{n}}_{\boldsymbol{m}})(x^{\boldsymbol{n}}_{\boldsymbol{m}} \,\varepsilon\, x^{\boldsymbol{n}+1}_{\boldsymbol{k}} \,.\equiv.\, \text{---}x^{\boldsymbol{n}}_{\boldsymbol{m}}\text{---}))$, if

'$---x^{n}_{m}---$' is a sentential function of the one variable 'x^{n}_{m}' from the translational part of the meta-language, and 'a' is taken as the structural description of the abstract '$x^{n}_{m} \ni ---x^{n}_{m}---$'.

We might be tempted to write *DesR1* more simply as

(1) $$\vdash a \text{ Des } x^{n}_{m} \ni ---x^{n}_{m}---,$$

where (as required). But the expressions asserted to be theorems by (1) are meaningless, because of the method here of handling abstracts. We are not regarding abstracts as substituends for variables, and hence we are not free to replace a variable by an abstract (of appropriate type). If we are to insert an appropriate abstract in place of 'x^{n}_{m}' here, this must be justified by a definition. We could, if desired, define expressions of the form given in (1), taking the corresponding expressions stipulated to be axioms by *DesR1* as definientia. But this is not needed for present purposes.

The second Rule of Designation is a restrictive rule, stating that if an expression a designates a class $x^{n}_{m}{}^{+1}$ then a is a predicate constant of type $n+1$.

DesR2. $\vdash a \text{ Des } x^{n+1}_{m} . \supset . \text{PredCon}^{n+1}\, a$.

The third Rule of Designation is a Principle of Uniqueness within any one type.

DesR3. $\vdash a \text{ Des } x^{n+1}_{j} . a \text{ Des } x^{n+1}_{k} :\supset: x^{n+1}_{j} = x^{n+1}_{k}$, for j and k of unequal length.

We need not assume separately a Uniqueness Principle as between types, for it can be proved that

$$\vdash a \text{ Des } x^{n+1}_{j} . \supset . \sim a \text{ Des } x^{m+1}_{k},$$

where n and m are strings of superscripts of unequal length.[1] For suppose that

$$\text{'}a \text{ Des } x^{n+1}_{j}\text{' and '}a \text{ Des } x^{m+1}_{k}\text{'}$$

both hold, for m and n of unequal length. Then by *DesR2* we should know that '$\text{PredCon}^{n+1}\, a$' and '$\text{PredCon}^{m+1}\, a$' both hold. This would be a contradiction of syntax. Hence

TB1. $\vdash \sim(a \text{ Des } x^{n+1}_{j} . a \text{ Des } x^{m+1}_{k})$, for m and n of unequal length.

The designation rules *DesR1* and *DesR3* together give the effect of Principles of Univocality. *DesR1* stipulates that any expression of a certain kind designates a class of such and such a type. *DesR2*

[1] This remark constitutes an improvement in the author's paper cited in the previous footnote, where a Uniqueness Principle as to type was erroneously thought needed as an axiom.

stipulates that it can designate at most one class of that type. Hence, using also *SynR4*, we have that

TB2. $\vdash \text{PredCon}^{n+1}\ a\ .\supset.\ (\text{E}x_m^{n+1})(a \text{ Des } x_m^{n+1}\ .\ (x_j^{n+1})(a \text{ Des } x_j^{n+1}\ .\supset.\ x_j^{n+1} = x_m^{n+1})$, if j and m are of unequal length.

Truth-definitions in terms of designation may be given as follows. Taking 'PNF' as in (VI,E), we let

'Tr a' abbreviate '(Sent a . (Eb)(Ec)(b PNF a . c R_{rp}^{invep} b . (Ex_1^2) (c Des x_1^2 . (x_1^1) $x_1^1 \varepsilon x_1^2$))'.

A sentence a is true, roughly speaking, if and only if the abstract formed from its prenex normal form (in the sense of (VI,E)) designates the universal class of objects of lowest type. Or we can define

'Tr a' as '(Sent a . (Eb)(Vbl' b . (Ex_1^2)(($b \frown invep \frown a$) Des x_1^2 . (x_1^1)$x_1^1 \varepsilon x_1^2$)))'.

According to this definition, a sentence a is true if and only if any vacuous abstract one forms from it, by using as the variable of abstraction a variable of first type, designates the universal class of individuals. There are other allied definitions also which may be given.

The proofs of adequacy of these various definitions are similar to those given above.

Note that here, as in SM_D^T, a very restricted semantical primitive will suffice for the purposes of a truth-definition. We could in fact allow

$$\text{'}a \text{ Des } x_m^{n+1}\text{'}$$

to be meaningful only where n is of length 1.

The relative consistency of the meta-language based on designation will in effect be shown in § D below.

With only slight changes one can use the method of designation to provide a semantics for T_ω or for a language formalized upon the basis of a logic of any finite order.

C. Determination in T. Another alternative semantical meta-language for T is gained by taking as the primitive semantical relation a restricted form of *determination* in place of designation.[1] Determination like designation is a semantical relation holding between expressions of a certain kind and classes of proper type. But the expressions here consist of sentential functions of at most one variable rather than predicate constants. There is therefore only a slight difference between these two semantical relations. In the semantical

[1] See R. Carnap, *Introduction to Semantics*, pp. 44–8.

meta-language based on determination, where 'Det' symbolizes this relation, forms such as

$$\text{'}x^{n}_{m} \text{ Det } a\text{'}, \text{ '}a \text{ Det } b\text{'}, \text{ and '}x^{n}_{m} \text{ Det } x^{k}_{q}\text{'}$$

are of course meaningless, and

$$\text{'}a \text{ Det } x^{n+1}_{m}\text{'}$$

is to be read 'the expression a determines the class x^{n+1}_{m}'.

The semantical meta-language here is very similar to that sketched in the preceding section. But in place of the Rules of Designation, we need corresponding *Rules of Determination*. These are obviously as follows.

DetR1. $\vdash (\mathrm{E}x^{n+1}_{k})(a \text{ Det } x^{n+1}_{k} \,.\, (x^{n}_{m})(x^{n}_{m} \,\varepsilon\, x^{n+1}_{k} \,.\equiv.\text{---}x^{n}_{m}\text{---}))$, if '$a$' is taken as the structural description of '$\text{---}x^{n}_{m}\text{---}$', a sentential function of the one variable 'x^{n}_{m}' from the translational part of the meta-language.

DetR2. $\vdash a \text{ Det } x^{n+1}_{m} \,.\supset.\, (\mathrm{E}b)(\text{SentFuncOne } a,b \,.\, \text{Vbl}^{n}\, b)$.

DetR3. $\vdash a \text{ Det } x^{n+1}_{j} \,.\, a \text{ Det } x^{n+1}_{k} :\supset: x^{n+1}_{j} = x^{n+1}_{k}$, for j and k of unequal length.

DetR3 is a form of a Uniqueness Principle within any one type. A Uniqueness Principle as between types is provable, essentially as in the semantical meta-language based on designation.

The truth-definitions for T based on determination closely resemble those based on designation, and may be omitted. Also the proof of the relative consistency of the meta-language based on determination is similar to that for the meta-language based on designation, to be given in a moment.

Clearly 'Des ' and 'Det ' are interdefinable in the appropriate meta-languages.

'a Des x^{n+1}_{m}' may abbreviate '$(\mathrm{E}b)(\mathrm{E}c)(\text{SentFuncOne } b,c \,.\, a = (c \frown \mathit{invep} \frown b) \,.\, b \text{ Det } x^{n+1}_{m})$',

and

'a Det x^{n+1}_{m}' may abbreviate '$(\mathrm{E}b)(\mathrm{E}c)(\text{SentFuncOne } a,c \,.\, b = (c \frown \mathit{invep} \frown a) \,.\, b \text{ Des } x^{n+1}_{m})$'.

Then *DesR1–DesR3* are provable from *DetR1–DetR3* in the meta-language based on 'Det', and conversely in that based on 'Des'. Therefore these two meta-languages for T are equivalent in an appropriate sense.

D. Concerning Designation and Determination in S. We might now try to construct a meta-language based on designation or determination for the set-theoretical language S, formulated above

in (VI,A), on the supposition that a truth-definition could be given similar to that for T. We might reasonably expect the truth-definition here to differ from that for T primarily in the treatment of variables. In particular, there would be here no distinction of type. But reflection for a moment upon the definientia of the truth-definitions for T in § B above, will show that a truth-definition for S in terms of designation, if possible at all, must be of a very different sort. Note that the definientia there contain a conjunct

$$'(x_1^1)x_1^1 \varepsilon x_1^2'$$

in an essential way. We should then expect here the purported definiens of a truth-definition for S to contain a conjunct of the form

$$'(\mathrm{E}y)(\text{-------} \,.\, (x)x \varepsilon y)'.$$

But within S there is no universal set y such that all sets are members of it. Hence our purported definiens would have the consequence that no sentence of S whatsoever could be true! For, by the purported definition, if a is a true sentence, then such a universal set exists. Similar remarks apply to a truth-definition for S based on determination. It is thus not clear that a truth-definition for S similar to those above can be given within a first-order meta-language based upon the narrow designation or determination relations being discussed.

The relations of designation and determination seem to be limited in a way in which denotation and satisfaction are not. In terms of either denotation or satisfaction, a semantical meta-language can be constructed for any first-order language-system in either the narrow or wider sense. The methods of designation and determination seem most appropriate for systems such as T, as we have seen. The problem of constructing a semantics for systems such as S on the basis of designation or determination is a somewhat special one which need not concern us further here.[1]

For systems such as T, the meta-language based on designation or determination can be shown equivalent to that based on denotation or satisfaction. In the appropriate meta-languages 'Sat' for T may be defined in terms of 'Det' for T, and conversely, as follows. In the meta-language based on 'Det'

'x_m^n Sat a' may abbreviate '$(\mathrm{E}x_1^{n+1})(x_m^n \varepsilon x_1^{n+1} \,.\, a \text{ Det } x_1^{n+1})$',

and

'a Det x_m^{n+1}' may abbreviate '$((\mathrm{E}b)(\mathrm{Vbl}^n\, b \,.\, \mathrm{SentFuncOne}\, a,b) \,.\, (x_1^n)(x_1^n \varepsilon x_m^{n+1} \,.\equiv.\, x_1^n \text{ Sat } a))$'

[1] Cf., however, G. D. W. Berry, "On Formalizing Semantics", abstract in *The Journal of Symbolic Logic*, 7 (1942), pp. 47–8.

in the meta-language based on 'Sat'. On the basis of these definitions one can show that *SatR1–SatR2* are provable from *DetR1–DetR3* in the meta-language for T based on determination, and conversely in that based on satisfaction. Similarly,

'a Den x_m^n' can abbreviate '$(\mathrm{E}x_1^{n+1})(a \text{ Des } x_1^{n+1} \,.\, x_m^n \,\varepsilon\, x_1^{n+1})$'

within the meta-language based on 'Des', and

'a Des x_m^{n+1}' can abbreviate '$(\mathrm{PredCon}^{n+1}\, a \,.\, (x_1^n)(x_1^n \,\varepsilon\, x_m^{n+1} \,.\equiv.\, a \text{ Den } x_1^n))$'

within SM_D^T. Using these definitions one can prove *DenR1–DenR2* from *DesR1–DesR3* within the meta-language based on 'Des', and conversely within SM_D^T. But 'Sat' and 'Den' we know to be interdefinable for all L, and 'Des' and 'Det' to be interdefinable for systems such as T. Hence the meta-languages based on satisfaction or denotation are equivalent with those based on determination or designation, taking T as the object-language. Hence also the meta-languages based on 'Des' or 'Det' for T are consistent if the joint language consisting of M and the translation of T is.

E. The Definition of Semantical Relations. Each of the semantical meta-languages developed above (including those in Chapters IV–VI) is determined by a certain semantical relation taken as primitive. Any such kind of semantics may be called a semantics *based on a special primitive*. There are also other general methods for constructing semantical meta-languages. We can in fact distinguish four such,

(1) the method of *definition*,
(2) the *truth-as-primitive* method,
(3) the method of *special primitives*,

and

(4) *non-translational* method.

In the preceding chapters and in §§ A–D above, we have had a fairly full account of (3), and in the succeeding chapters we shall have a detailed presentation of (4). In the present section, for the sake of completeness, we wish to consider very briefly the methods (2) and (1).

A semantics of the kind (2), in which a predicate 'Tr' is taken as the sole semantical primitive, can easily be formulated. The semantical axioms would be provided by a Rule of Adequacy together with the law that any truth (of L) is a sentence (of L).[1] If L contains arithmetic, the underlying syntax can either be arithmetized or

[1] See Tarski, *Der Wahrheitsbegriff*, § 5.

developed in the manner of Chapter III. The resulting semantical meta-language would be closely akin to those already discussed.

Of course 'Tr' itself might be regarded as a special primitive, and hence the method (2) might appear as a special case of (3). 'Tr' being so fundamental a notion, however, we may regard (2) as a distinct method.

Let us now turn to the method (1).[1] Here we allow *no* semantical primitives but require that *all semantical terms and relations be defined*. To define the notions of semantics we must presuppose in the meta-language some devices at least as powerful as those in the meta-languages already formulated. In fact, as we have already noted in (V,H), the semantical meta-language must always be "essentially richer" in important respects than the object-language. If we have no specific semantical primitive or axioms characterizing it, the essential richness of the meta-language must be found elsewhere. We can in fact gain the desired power in the meta-language by requiring that it be a language of order *higher* than that of the object-language. If L^n is an object-language of order n, we can formulate a semantical meta-language of order $(n+1)$ in which the basic notions of semantics (relativized to L^n) are definable. To see this more clearly, let us take L^n as a first-order language (for $n = 1$) and sketch briefly its semantics within a meta-language of *second* order.

Let L be as in Chapter III. Within the second-order semantical meta-language, which we shall call SM_2^L, we presuppose the syntax of Chapter III. Likewise L is translated within SM_2^L. In addition SM_2^L contains two kinds of variables of second type. These are to be thought of as ranging respectively over *classes* of objects (of L) or of expressions (of L) or of both, or over *dyadic relations* between objects and expressions, and between objects and objects, expressions and expressions, or expressions and objects. (We could also admit variables over triadic relations but these will not be needed.) Let 'F' and 'G' with or without numerical subscripts be the class variables, and let 'R' and 'Q' with or without subscripts be the dyadic relational variables.[2] Within SM_2^L, in addition to the axioms of syntax and the non-logical axioms of L, one has also the logical axioms concerning the truth-functions and quantifiers, covering quantifiers upon all kinds of variables admitted. (See (VI,G).) The salient features of SM_2^L should be clear from this brief sketch.

We may now define within SM_2^L either 'Den' or 'Sat'. Let us choose 'Sat'. Two or three preliminary notions are needed. We let

[1] This method is due primarily to Tarski, in *Der Wahrheitsbegriff*.

[2] Strictly the class variables are not needed here either and may be dropped if desired.

'OddTildeString a' abbreviate '(Eb)(a FrIng b . (c)(c FrIng b :⊃: c = *tilde* .v. (Ed)(d Pr_b c. c = (*tilde*∩*tilde*∩d)))',

and similarly,

'EvTildeString a' abbreviates '((b)(PS b . b Seg a :⊃: b = *tilde*). ∼ OddTildeString a)'.

These definitions define respectively the notions of being an *odd* string of tildes and an *even* string of tildes. To simplify, suppose for the moment that L contains no primitive '=' and only *two* primitive predicate constants, 'P' and 'P′', and that these are respectively of one and two places. We can then define a notion of *pseudo-atomic satisfaction* as follows, by a kind of enumeration.

'x PsAtSat a' abbreviates '((Eb)(Vbl b . a = (*pee*∩b) . Px) .v. (Eb)(Ec)(Ed)(Ee)(Vbl b . Vbl c . EvTildeString d . OddTildeString e : ((a = (b *qu pee*∩*ac*∩b∩c) .v. a = (b *qu* d∩*pee*∩*ac*∩b∩c)) . (y)P′yx) .v. (a = (b *qu* e∩*pee*∩*ac*∩b∩c) . (y)∼P′yx) .v. ((a = (c *qu pee*∩*ac*∩b∩c) .v. a = (c *qu* d∩*pee*∩*ac*∩b∩c)) . (y)P′xy) .v. (a = (c *qu* e∩*pee*∩*ac*∩b∩c) . (y)∼P′xy)))'.

Note that where x pseudo-atomically satisfies a, a is of one of the forms (where $\boldsymbol{x}$ and $\boldsymbol{y}$ are distinct variables)

'P$\boldsymbol{x}$',
'($\boldsymbol{y}$)P′$\boldsymbol{yx}$',
'($\boldsymbol{y}$)$\underbrace{\sim\ldots\sim}_{\text{even number of times}}$P′$\boldsymbol{yx}$',
'($\boldsymbol{y}$)$\underbrace{\sim\ldots\sim}_{\text{odd number of times}}$P′$\boldsymbol{yx}$',
'($\boldsymbol{y}$)P′$\boldsymbol{xy}$',
'($\boldsymbol{y}$)$\underbrace{\sim\ldots\sim}_{\text{even number of times}}$P′$\boldsymbol{xy}$',

or

'($\boldsymbol{y}$)$\underbrace{\sim\ldots\sim}_{\text{odd number of times}}$P′$\boldsymbol{xy}$'.

(If L contains additional primitive predicates or a primitive '=' this list must be suitably extended.) The notion of satisfaction may then be defined as follows.

'x Sat a' abbreviates '(R)((y)(b)(y R b :≡: y PsAtSat b .v. (Ed)(Ee)(SentFuncOne d,e . b = (*tilde* d) . ∼y R b) .v. (Ec)(Ed)(Ea′)(SentFuncOne c,a′ . SentFuncOne d,a′ . b = (c *vee* d) . (y R c .v. y R d)) .v. (Ec)(Ed)(SentFuncOne c,d . b = (d *qu* c) . (z)z R c)) .⊃. x R a)'.[1]

[1] This definition is a modification of that of Tarski, *op. cit.*, pp. 311–12.

On the basis of this definition, *SatR1* and *SatR2* (of § A above) may be proved within SM_2^L, for appropriate L. In terms of 'Sat' we know that the other notions in the semantics of L may be defined, and from *SatR1* and *SatR2* the basic theorems of semantics may be proved.

Note that this definition of 'Sat' is in effect a recursive definition. In the hypothesis of the definiens, we consider '$y\ R\ b$' first in the simplest cases where y pseudo-atomically satisfies b. Then, presupposing 'Sat' defined for such b, we go on to cases of greater complexity, involving '$\sim$', 'v', or quantifiers. The effect of the recursive procedure here is achieved by the quantifier '(R)', the presence of which is essential in the definiens.

If the object-language L is taken as a language-system of some finite order, a definitional semantical meta-language for L of next higher order may be constructed in essentially the same way. Also this definitional method can be extended to apply to very powerful object-languages such as T or T_ω or S. For a definitional semantics of such systems we either presuppose in the meta-language variables of *transfinite type* or use other mathematical procedures of roughly commensurate power. But some doubts may well arise concerning the legitimacy of using such procedures in semantics. And anyhow the problem of gaining a definitional semantics for such powerful systems is somewhat special and need not concern us further here.[1]

We have noted that a definitional semantical meta-language of an object-language L_1 must be "essentially richer" in logical or semantical modes of expression than L_1 itself. Often the semantical meta-language is *materially richer* than L_1 in the sense of containing not only a translation of L_1 but some suitable material extension of L_1 as well. In this case, the meta-language may contain, roughly speaking, some L_2 of which a translation of L_1 is a part, or it may contain more powerful axioms than those of L_1. The primitives of L_2 should be either such as to contain all those of L_1 or such that the primitives of L_1 are definable in terms of them. And the axioms of L_2 should be such that all theorems of L_2 are provable from them, but not conversely. Of course materially richer meta-languages are *a fortiori* essentially richer also. We shall have little occasion here to speak further of such meta-languages, but they may be useful for some purposes and it is important to distinguish them from the kinds we have been considering.

F. In Favor of SM_D^L and Allied Semantical Meta-Languages. In closing this part of the discussion, and before going on to

[1] See, however, the papers by Kemeny and Wang cited in footnote 1, p. 144.

non-translational semantics, let us review some of the general features of the semantical meta-languages based on special primitives, and compare and contrast these with meta-languages such as SM_2^L based on the method of definition.

In the first place, a meta-language based on a special primitive provides a direct axiomatic treatment of semantics, the axioms characterizing implicitly the special primitive adopted. A semantics based on a special primitive is in some respects analogous to a direct axiomatization of arithmetic, in which natural numbers (or positive integers) are taken as values for variables. For some purposes one would wish a direct axiomatic characterization of the natural numbers rather than a logical analysis of them such as that given by Frege or Russell. Here a direct axiomatic characterization of multiple denotation, e.g., is given rather than a logical analysis of it using the definitional method. Both kinds of semantical methods should be of interest and useful for some purposes, just as both analogous methods are of interest in formulating arithmetic. The direct axiomatic characterization has advantages in both cases.

The semantical meta-languages based on special primitives are in accord with the general philosophical view of first-order constructivism. Given an object-language L, we wish to provide the basic notions in the semantics of L using only very limited logical and semantical notions. The reasons for preferring such limited notions should be evident enough. We shall consider in detail some of the reasons in favor of first-order constructivism in a later chapter (XIII). Here we note again that if L is taken as a first-order language in the narrow sense, then SM_D^L and the allied semantical meta-languages are likewise. But if L is a first-order language only in the wider sense, then SM_D^L and the allied semantical meta-languages are also. The character of the semantical meta-languages as first-order languages depends solely upon the character of the object-language, and in no way upon the syntactical and semantical notions being used. On the other hand, if the object-language is of order n, a semantics based on the method of definition can be formulated only within a system of at least order $(n+1)$ (or of order n only in an extended sense). A semantics formulated in terms of a special primitive has the advantage of presupposing only the extremely simple kind of basic logic discussed in Chapter II, for first-order L.

We have noted also that the semantical method of special primitives is general in the sense of being applicable to all object-languages of the appropriate kind, including such languages as S and T. This does not mean of course that the object-languages are taken as values for variables. (See (XIII,H).) The methods here are general

rather in the sense that given any fixed L, a method of constructing a semantical meta-language of L is immediately forthcoming. (A possible exception to this is the case of semantical meta-languages based on designation or determination for object-languages such as S.) We have noted above certain limitations in this respect of the semantical method of definition, unless suitably extended.

We must note also the extreme simplicity of the meta-languages here. All the primitives used are especially simple and perspicuous semantical relations. No complicated concepts are required in constructing the theories of the special primitives adopted. (In particular, no natural numbers are required as values for variables, no infinite sequences of entities of any kind, no expressions of transfinite type, etc.)[1] The logical notions used in a definitional semantics are roughly tantamount to those required for developing the mathematical theory of real numbers from a foundation in integers.[2] It might seem somewhat anomalous to require such powerful notions to formulate semantical theory, which is and should be essentially simple in structure.

These various advantages also accrue to a semantics with 'Tr' taken as a primitive.

On the other hand, a semantics based on the method of definition has the important advantage of having all semantical terms definable, but at the expense of presupposing a basic logic of greater power.

In a deeper sense, however, a semantics based on 'Tr' or a special primitive is fundamentally like a definitional semantics in the following respect. For both kinds of theories we need new primitives. Either we take new logical primitives or what we call specifically *semantical* ones. But whether we call them this or that is of little import. In the former case, we take in effect a multiplicity of primitives, variables and quantifiers over classes, dyadic relations, and so on, of type at least one higher than that of the variables of highest type occurring in the object-language. New axioms and rules are needed to characterize these new quantifiers. And similarly in a semantics based upon a special primitive. But here we need just one new primitive characterized by suitable semantical rules, *chosen just to provide for semantics and no more.* In the definitional method the new quantifiers and rules provide for a great deal more than is intended. In terms of such quantifiers various new areas of logical

[1] Some such entities are required fundamentally in Tarski's treatment, although not within meta-languages based upon the method of definition such as SM_2^L discussed above.

[2] Cf. Hilbert and Bernays, *op. cit.*, vol. 2, Supp. IV.

theory can often be accommodated. But if such quantifiers are not needed and in fact never used, there seems little reason to introduce them for semantical purposes. A semantics based upon a special primitive concentrates attention upon the notions actually needed for the purpose at hand.

We could see this a little more clearly if we were to formulate a semantics based not on a special primitive but upon just the new variables, quantifiers, and *contexts in which they occur* which are actually used. We have seen from § E that we need only new variables and quantifiers over dyadic relations (of a certain kind). If we were to develop the semantics of § E in detail we should see, furthermore, that we need these quantifiers only as prefixed to formulae *of certain forms*. Thus even quantifiers over all dyadic relations as prefixed to all admissible formulae are not needed in semantics. (We need not show this in detail; it should be clear enough from the considerations of § E.) In effect, in a semantics based on a special primitive, we pick out just the quantifiers and contexts to which they are prefixed that are actually needed. The special primitive enables us to symbolize these quantifiers and contexts in a simple, convenient, and perspicuous way.

There seems thus to be no genuine quarrel between these two types of semantics. The one is in effect a mere fragment of the other and hence enjoys many advantages. Because heretofore attention has been focussed almost exclusively upon definitional semantics, we have thought it of interest in this book to develop the method of special primitives in considerable detail.

CHAPTER VIII

NON-TRANSLATIONAL SEMANTICS (I)

THE various semantical meta-languages developed in the preceding chapters have one important feature in common, namely, that they contain in one way or another the whole of the object-language L to which they are applied. One way of achieving this is to require, as we have seen, that each symbol of the object-language have a corresponding translation within the semantical meta-language. Or we can think of the semantical meta-language as simply an extension of the object-language, achieved by the addition of certain syntactical and semantical primitives together with axioms characterizing them. But in either case, the object-language L reappears *in toto* within the semantical meta-language. We may speak of any such semantical meta-language as a *translational* meta-language, and the semantics involved as a *translational semantics*. All semantical meta-languages which have heretofore been studied in any technical detail in this book or elsewhere, seem to be translational in character.

In this chapter the foundations of a semantical theory will be laid which, because it lacks this crucial feature, we shall call a *non-translational semantics*, and the meta-languages to be formulated we shall call *non-translational, semantical meta-languages*. These meta-languages afford a radical departure from the kinds of meta-languages studied heretofore. In some respects they are less powerful, as we shall see, and certain principles of translational semantics will have to be sacrificed. Bearing in mind the vital role these principles have played there, it is perhaps somewhat remarkable that so much can be accomplished without them. But this is anticipating.

Multiple denotation, and the other semantical primitives considered in Chapter VII, it will be remembered, relate the expressions of a language to the objects about which the language speaks. A semantics based upon such relations must thus contain expressions for the symbols of the object-language as well as for its objects. The proposal of the present chapter is to drop any such relation as a primitive and in its place to take either *subsumption* or *comprehension* (as defined in Chapter IV). If we then introduce no further primitives, we gain a semantical meta-language which speaks exclusively of the expressions of the object-language and not of its

objects, because the relata of the comprehension and subsumption relations are exclusively expressions. In taking such a purely linguistic relation as a primitive we avoid the necessity of augmenting the meta-language by incorporating the object-language within it. The attendant gain in economy and simplicity is of course enormous.

Several alternative non-translational meta-languages will be formulated, and some of these will be simpler in important respects than others. These various meta-languages will be shown equivalent and later (in (XIII,F)) will be shown to provide a *minimal* foundation for semantics in an appropriate sense.

In § A the *primitive framework* of a semantical meta-language based on comprehension will be discussed. Some preliminary *definitions* will be given in § B. In § C the semantical *truth-concept* will be defined in several alternative ways, and in § D *rules* providing one formulation of non-translational semantics will be given. Some *alternative formulations* are suggested in § E. The exposition is continued in the next chapter.

A. Primitives. The first non-translational semantical meta-language to be discussed consists of the elementary concatenation theory of Chapter III, augmented by a theory of comprehension. The basic logic presupposed is of the kind considered in Chapter II. The meta-language is thus of an extremely simple structure. It is a simple, applied language-system of first order, with its fundamental domain taken as individuals in a very strict sense. Where 'Cmprh' is the sole semantical primitive, the meta-language will be called SM_C^L, the semantical meta-language for L based on comprehension. The only variables needed are the expressional variables 'a', 'b', etc.

The notion of a *term* of SM_C^L can be defined (recursively) within a meta-language of SM_C^L as any variable or any structural-descriptive constant, or as '$(\boldsymbol{x} \frown \boldsymbol{y})$', where $\boldsymbol{x}$ and $\boldsymbol{y}$ are any terms. The notion of *formula* is definable as any expression '$\boldsymbol{x} = \boldsymbol{y}$' or '$\boldsymbol{x}$ Cmprh $\boldsymbol{y}$', where $\boldsymbol{x}$ and $\boldsymbol{y}$ are terms, or as '$\sim \boldsymbol{A}$', '$(\boldsymbol{A} \vee \boldsymbol{B})$', '$(\boldsymbol{x})\boldsymbol{A}$', or '$\boldsymbol{x} \ni \boldsymbol{A}\ \boldsymbol{y}$', where $\boldsymbol{A}$ and $\boldsymbol{B}$ are formulae, $\boldsymbol{x}$ a variable, and $\boldsymbol{y}$ a term. This gives in summary a complete description of the primitive framework of SM_C^L. The object-language L here is to be taken in the sense of Chapter III, i.e., as any simple, applied, language-system of first order with identity but containing no individual or functional constants.

B. Some Definitions. Before giving the rules of SM_C^L, let us glance at some definitions of further notions which will be useful. The definitions of the various *syntactical* notions in Chapters III–V, are to be regarded as reappearing here as definitions of SM_C^L.

Having taken comprehension as a primitive, we might have occasion to make use of its converse. To justify this we need the definition of *subsumption*, as in (IV,C).

'*a* Sub *b*' abbreviates '*b* Cmprh *a*'.

The next definition is that of the notion of a *denoting expression*. There was no formal need for this notion above (in Chapters IV–VII). Denoting expressions were there identified with predicate constants. Here also they will be identified with predicate constants, but explicitly by means of a rule. But first this notion must be defined in terms of comprehension. An expression *a* of *L* is a denoting expression of *L* if and only if there is some expression *b* of *L* such that either *a* comprehends *b* or *b* comprehends *a*, in other words, if and only if *a* is a member of the field of the comprehension relation. We let

'DE *a*' abbreviate '(E*b*)(*a* Cmprh *b* .v. *b* Cmprh *a*)'.

Of course this definition is not very informative. We still do not know which expressions of *L* are denoting expressions and which are not. We shall postulate below that the denoting expressions of *L* are just the predicate constants. (See the first rule in § D.) Let us refer to this assumption as the *Rule of Denoting Expressions*. Note that 'predicate constant' is a purely syntactical phrase, whereas 'denoting expression' is semantical, being defined in terms of comprehension. The Rule of Denoting Expressions interrelates these two notions.

Two expressions are *co-extensive* (as in (IV,C)) if they mutually comprehend one another.

'*a* CoExt *b*' abbreviates '(*a* Cmprh *b* . *b* Cmprh *a*)'.

The following definitions also are closely allied with the corresponding definitions of (IV,C). The definitions there are in terms of multiple denotation. Here the notions involved must be defined in a new way in terms of comprehension.

An expression is semantically *null* if and only if it is comprehended by every predicate constant.

'Null *a*' abbreviates '(*b*)(PredCon *b* .⊃. *b* Cmprh *a*)'.

From syntax it follows that there exists at least one predicate constant and hence where *a* is null there will be at least one *b* which comprehends *a*. Hence by the Rule of Denoting Expressions, if *a* is null *a* is a predicate constant, and there is no need to add to the definiens of this definition the clause 'PredCon *a*'. As one would expect from the considerations in Chapters IV and V, the theory allows for an infinity of distinct but co-extensive predicate

constants. Intuitively speaking, again in terms of multiple denotation, null predicate constants are those which denote nothing.

In an analogous way we shall also speak of semantically *universal* expressions.

'Univ a' abbreviates '(b)(PredCon b .⊃. a Cmprh b)'.

All universal expressions will be predicate constants for reasons similar to those in the case of null predicate constants so that there is no need of adding 'PredCon a' to the definiens. The postulates will be such as to assure the existence of an infinity of distinct but mutually co-extensive universal predicate constants. Intuitively speaking, a universal predicate constant denotes all objects of L.

It is useful to have a special notation for non-null predicate constants. Therefore as in (IV,C) we let

'E!a' abbreviate '(PredCon a . ~Null a)'.

Suppose we have a non-null predicate constant a which is such that it is comprehended by every non-null expression which it comprehends. Such a predicate constant a denotes (1) at least one object, and (2) at most one object. That (1) holds is evident because a is non-null. That (2) holds also may be seen as follows. (From syntax we know that there is an infinity of predicate constants available in L.) Let b be any non-null expression which a comprehends. Suppose it so happens that b denotes just two objects. But b also comprehends a on the supposition, so that a also can denote at most two objects. Let c be one of the multiplicity of expressions which denote just one object. If a comprehends c then, by the supposition, c must also comprehend a so that a can then denote at most one object. Therefore if a is a non-null predicate constant and is comprehended by every non-null expression which it comprehends, then a denotes one and only one object. Such an expression is a semantically *unit* predicate constant.

'Unit a' abbreviates '(E!a . (b)(~Null b . a Cmprh b :⊃: b Cmprh a))'.

A *common* predicate constant is one which is not unit and is non-null.

'Com a' abbreviates '(E!a . ~Unit a)'.

The *semantical sum* of two predicate constants b and c denotes everything denoted by either b or c. Expressed wholly in terms of comprehension, we can say that an expression a is a semantical sum of the expressions b and c where a comprehends both b and c and where every d which comprehends both b and c also comprehends a.

'*a* Sum *b*,*c*' abbreviates '(*a* Cmprh *b* . *a* Cmprh *c* . (*d*) (*d* Cmprh *b* . *d* Cmprh *c* :⊃: *d* Cmprh *a*))'.

It will be postulated in effect that, given any predicate constants *b* and *c*, at least one semantical sum of *b* and *c* exists. It will then be easy to show that semantical sums possess the appropriate formal properties.

In a similar way, an expression *a* is said to be a *semantical product* of expressions *b* and *c* if and only if both *b* and *c* comprehend *a* and every *d* which both *b* and *c* comprehend is also comprehended by *a*.

'*a* Prod *b*,*c*' abbreviates '(*b* Cmprh *a* . *c* Cmprh *a* . (*d*)(*b* Cmprh *d* . *c* Cmprh *d* :⊃: *a* Cmprh *d*))'.

The product of two predicate constants denotes just those objects denoted by both. The existence of at least one semantical product of any two predicate constants will be postulated, and the appropriate formal laws concerning these products will be provable.

Finally, we can say that an expression *a* is a *semantical negative* of *b* if and only if, for every *c*, both *a* and *b* comprehend *c* if and only if *c* is null, and for every *c*, *c* comprehends both *a* and *b* if and only if *c* is universal.

'*a* Neg *b*' abbreviates '((*c*)(*a* Cmprh *c* . *b* Cmprh *c* :≡: Null *c*) . (*c*) (*c* Cmprh *a* . *c* Cmprh *b* :≡: Univ *c*))'.

Again, the existence of at least one semantical negative of any predicate constant will be postulated, and the formal laws concerning these negatives will be provable.

It is a little difficult at this point to state why these notions are useful within non-translational semantics. In particular, they help to compensate for the sparse modes of expression available. The notion of *Sum* will help us to handle sentences of the object-language of the form of material disjunctions, that of *Prod*, of conjunctions, and that of *Neg*, negations. Even more important than these is the notion of the *semantical implexion* of predicate constants, which will be introduced in the next chapter.

C. Truth. We are now in a position to define the semantical truth-concept for *L* within SM^L_C. This can be done in any of the four ways discussed in (V,A and B). In fact these four definitions may now be regarded as definitions of SM^L_C. The definiens in each of these definitions contains 'Univ' in an essential way, which must now be regarded as defined in terms of comprehension as above, rather than in terms of multiple denotation as in (IV,C).

One of the most important features of non-translational semantics

is that the requirement of adequacy for the truth-definition must be abandoned. The reason is that it cannot even be stated for the formalism SM_G^L, because SM_G^L contains only structural descriptions, not translations, of the expressions of L. To state the requirement of adequacy presupposes that within the semantical meta-language one have both, i.e., that one can both mention and use the expressions of the object-language, as we have noted in (V,C).

In a certain sense, the requirement of adequacy for the truth-definitions here must be abandoned. But they are clearly *in accord with* it. In other words, the meaning given to the primitive 'Cmprh' is such that the resulting truth-concept is essentially the same as one within a formalism in which the requirement of adequacy can be stated and proved in full generality. It is not that adequacy is in any way denied here or its negative assumed. That the truth-concepts defined are clearly in accord with it cannot be explicitly proved, but must rest rather on intuitive grounds concerning the precise meaning of 'Cmprh'. (See (XII,G) below.)

For purposes of the subsequent development in this chapter and in Chapters IX and X we shall assume that the truth-concept is defined by the fourth of these definitions, corresponding to (V) of (V,B), in terms of vacuous abstracts. The other truth-definitions will not be referred to subsequently, not being needed for present purposes. (The various truth-definitions are, however, equivalent to each other, although a proof of this will not be given. But see *TD17* in Chapter IX.)

D. General Rules of Comprehension. The rules of SM_G^L fall into three groups, (i) logical rules, (ii) syntactical rules, and (iii) semantical rules. The logical rules are those given above in (III,B), providing for a simple, applied, language-system of first order with identity. The syntactical rules comprise *SynR1–SynR3* of (III,B), but we drop the Rule of Infinite Induction, *SynR4*. In its place we take two of its consequences, the Rule of Framed Ingredients, *TG4*(III), and the Rule of Expressions, *TG13*(III). (Using these, it will be recalled, we can prove the Induction Rule *TG14*, and therewith the various theorems of (III,C).) By dropping *SynR4* we do not appreciably weaken the syntax of L, and this rule is not needed for semantical purposes here.

The semantical rules of SM_G^L, the *General Rules of Comprehension*, stipulate properties of predicate constants which in no way depend upon the specific character of the object-language. They are general in the sense that they apply to all languages alike of the kind under discussion. They do not state which predicate constants of the

object-language comprehend which. Of course some predicate constants of L will comprehend certain predicate constants wholly in virtue of the underlying logic. This will follow from the General Rules. But properties of comprehension which in any way depend upon or reflect the character of the non-logical axioms or theorems of the object-language cannot be established merely by using the General Rules. These must be provided in some other way. (See (IX,E).)

The General Rules of Comprehension are as follows. The first, the *Rule of Denoting Expressions*, has already been mentioned above.

GRC1. $\vdash$ DE a .≡. PredCon a.

We also have a *Transitivity Rule.*

GRC2. $\vdash$ a Cmprh b . c Cmprh a :⊃: c Cmprh b.

It has been mentioned that given any two predicate constants a and b, there must exist at least one semantical sum of a and b. Rather than to postulate this directly, it will be assumed that, where a, b, and c are expressions interrelated in a certain way syntactically, a is a semantical sum of b and c. Suppose that there is a b', a c' and a d such that b' is a sentential function of the one variable d, b is $(d \frown invep \frown b')$, c' is a sentential function of one variable d, c is $(d \frown invep \frown c')$, and a is $(d \frown invep \frown (b' \ vee \ c'))$. Under these circumstances we stipulate that a be a semantical sum of b and c. This is provided by the *Sum Rule.*

GRC3. $\vdash$ (Eb')(Ec')(Ed)(SentFuncOne b',d . SentFuncOne c', d . $b = (d \frown invep \frown b')$. $c = (d \frown invep \frown c')$. $a = (d \frown invep \frown (b' \ vee \ c')))$.⊃. a Sum b,c.

By means of this rule we gain a certain correspondence between the notions embodied in '*vee*' and 'Sum'.

A similar principle is postulated concerning semantical products. Recall that '(*a dot b*)' was defined in (III,D) as '*tilde* (*tilde a vee tilde b*)'. The correspondence required between the notion contained in '*dot*' and 'Prod' is similar to that between '*vee*' and 'Sum', and is given by the following *Product Rule.*

GRC4. $\vdash$ (Eb')(Ec')(Ed)(SentFuncOne b',d . SentFuncOne c',d . $b = (d \frown invep \frown b')$. $c = (d \frown invep \frown c')$. $a = (d \frown invep \frown (b' \ dot \ c')))$.⊃. a Prod b,c.

A *Negative Rule* is needed to give the circumstances under which one predicate constant is a semantical negative of another. This will give the correspondence desired between '*tilde*' and 'Neg'.

GRC5. $\vdash$ (Ec)(Ed)(SentFuncOne c,d . $a = (d \frown invep \frown c)$. $b = (d \frown invep \frown tilde \frown c))$.⊃. b Neg a.

Note that what are essentially the converses of the axioms stipulated by *GRC3*, *GRC4*, and *GRC5* do not hold. E.g., b can be a negative of a in the sense defined where b and a are not interrelated in the way required in the hypothesis of *GRC5*. '$x \ni \sim Px$' is a semantical negative of '$x \ni (Px \,.\, (Px \vee \sim Px))$', e.g., although these two abstracts are not interrelated in the way required by the hypothesis.

The next rule, the *Rule of Non-Null Semantical Products*, gives a sufficient condition under which a semantical product is non-null.

GRC6. $\vdash d \text{ Neg } b \,.\, c \text{ Prod } a,d \,.\, \sim b \text{ Cmprh } a :\supset: \sim\text{Null } c.$

This principle is needed for technical purposes, especially for the proof of a distributive law.

Also a rule concerning semantically null products with a common factor is needed, the *Rule of Semantically Null Products*.

GRC7. $\vdash d \text{ Sum } a,b \,.\, d \text{ Cmprh } c \,.\, (e)(c \text{ Cmprh } e \,.\, (a \text{ Cmprh } e \,.\vee.\, b \text{ Cmprh } e) :\supset: \text{Null } e) :\supset: \text{Null } c.$

This rule likewise is needed for technical reasons, mainly for the proof of a distributive law.

The next General Rule *GRC8* is of a somewhat different character. The preceding rules have stipulated certain general properties of comprehension.[1] The next states rather that certain predicate constants are universal. This rule is general because it holds for all languages L of the kind being investigated and in no way depends upon the specific axiomatic structure of L. The purpose of this rule is to enable us to show that any *logical axiom* of L (provided it be a sentence), or any closure of such, is true in the appropriate sense.

GRC8. $\vdash (\text{E}c)(a \text{ Clsr}' \, c \,.\, \text{LogAx } c) \,.\, \text{Vbl } b :\supset: \text{Univ } (b \frown invep \frown a).$

Also a rule is required in order to differentiate properly between universal and null predicate constants, the *Rule of Differentiation*.

GRC9. $\vdash \text{Univ } a \,.\, \text{Null } b :\supset: \sim b \text{ Cmprh } a.$

It follows from this rule that universal and null predicate constants are not co-extensive. Note that the rule does not stipulate that there are such things as universal or null predicate constants. This must be provided in some other way.

The definition of truth to be used in the subsequent development is in terms of vacuous abstracts. To fix explicitly the semantical properties of these, three *Rules of Vacuous Abstracts* are required. The first interrelates (possibly) vacuous with (possibly) non-vacuous abstracts in a certain way.

[1] *GRC2–GRC7* are in fact akin to some of Tarski's inclusional postulates for Boolean algebra. See his "Grundzüge der Boole'schen Algebra I", especially pp. 181–2.

GRC10. $\vdash$ SentFuncOne b,a . Vbl c . Univ $(a \frown invep \frown b) :\supset:$ Univ $(a \frown invep \frown c\ qu\ b)$.

What amounts essentially to the converse of *GRC10* is provable. (See (IX,D), *TD18*.)

The second rule stipulates in effect that any vacuous abstract $(b \frown invep \frown a)$ comprehends any vacuous abstract $(c \frown invep \frown a)$, for distinct variables b and c.

GRC11. $\vdash$ Sent a . Vbl b . Vbl c . $\sim b = c :\supset: (b \frown invep \frown a)$ Cmprh $(c \frown invep \frown a)$.

It is an immediate consequence of *GRC11* that the abstracts concerned are co-extensive.

The third Rule of Vacuous Abstracts stipulates that any vacuous abstract is either universal or null.

GRC12. $\vdash$ Sent a . Vbl $b :\supset:$ Univ $(b \frown invep \frown a)$ .v. Null $(b \frown invep \frown a)$.

By means of this rule we shall be able to prove that, given a sentence a of L, either a or its negative is true.

Also we need a rule closely related to *GRC11* but concerned with non-vacuous abstracts. This *Rule of Bound Abstract-Variables* amounts to a rule for rewriting bound variables in non-vacuous abstracts.

GRC13. $\vdash$ SentFuncOne a,b . b FV a . d SF^c_b a . $\sim b = c :\supset: (b \frown invep \frown a)$ Cmprh $(c \frown invep \frown d)$.

Finally, we assume that the primitive one-place predicate constants of L are co-extensive with certain abstracts. Taking L as in Chapter II (but with no primitive individual or functional constants), we recall that 'P' is the only one-place primitive predicate constant. Hence we need assume merely the following.

GRC14. $\vdash$ *pee* CoExt $(ex \frown invep \frown pee \frown ex)$.

This completes the list of the General Rules of Comprehension. The consistency of these rules relative to that of the logical and syntactical rules of SM^L_G will be proved in (IX,G).

If we were to define 'Cmprh' in SM^L_D as in (IV,C), as well as 'Sum', 'Prod', etc., we would note that analogues of *GRC2–GRC14* are in fact provable in SM^L_D. Within SM^L_D we did not introduce formally the notion of a *denoting expression*. This could have been done, however, by defining

'DE a' as 'PredCon a'.

Then the analogue of *GRC1* would follow immediately. These rules were not given in the discussion of SM^L_D, for they are there relatively

unimportant. There they form a mere fragment of a much richer theory. Now our task is to show that just these principles suffice for the semantical purposes at hand.

E. Some Alternative Bases. 'Cmprh' has been taken above as the sole semantical primitive for non-translational semantics. Several alternative primitives will now be considered.

Clearly 'Cmprh' is non-symmetrical, i.e., we can show, on the basis of *GRC1–GRC14* that

$$\vdash \sim(a)(b)(a \text{ Cmprh } b \,.\supset.\, b \text{ Cmprh } a).$$

For by *GRC9* we have that

$$\vdash \text{Univ } a \,.\, \text{Null } b \,.\, a \text{ Cmprh } b :\supset: \sim b \text{ Cmprh } a,$$

and hence

$$\vdash (\text{E}a)(\text{E}b)(a \text{ Cmprh } b \,.\, \sim b \text{ Cmprh } a),$$

provided

$$\vdash (\text{E}a)(\text{E}b)(\text{Univ } a \,.\, \text{Null } b \,.\, a \text{ Cmprh } b).$$

(See *TB6* in (IX,B) below.) Primitive two-place predicates which are symmetrical are often preferable to non-symmetrical ones.[1] Hence it is of interest to consider also symmetrical primitives of commensurate defining power. We need not seek far for such predicates. Let 'O' express the relation of *semantical overlapping*. 'a O b' holds where a and b are both predicate constants of L denoting (in the sense of multiple denotation) at least one object in common. Or, let '$\int$' express the relation of being *semantically discrete*, so that '$a \int b$' holds where a and b are both predicate constants of L denoting *no* object in common.[2] Both these predicates are symmetrical, and either may be taken as a primitive in place of 'Cmprh'.

In terms of 'O', 'Cmprh' is definable.

'a Cmprh b' may abbreviate '(PredCon a . PredCon b . $(c)(c \text{ O } b \,.\supset.\, c \text{ O } a)$)',

so that on the basis of 'O' all the notions of non-translational semantics are definable. Rather than merely replace phrases of the form 'a Cmprh b' by their definientia in terms of 'O' in the definitions of the preceding sections, we must seek other more direct definitions. And similarly, in the primitive expansions of *GRC1–GRC14*, we

[1] See N. Goodman, "The Logical Simplicity of Predicates", *The Journal of Symbolic Logic*, 14 (1949), pp. 32–41, and "An Improvement in the Theory of Simplicity", *ibid.*, pp. 228–9. Also *The Structure of Appearance*, pp. 59–75.

[2] The symbols for semantical overlapping and discreteness are Goodman's symbols for corresponding notions in the calculus of individuals. Cf. *The Structure of Appearance*, pp. 43–4.

should not now simply replace phrases of the form '*a* Cmprh *b*' by their definientia in terms of 'O'. Rather new rules must be sought directly in terms of 'O'. Similar remarks apply to a non-translational semantics based on semantical discreteness.

We have chosen 'Cmprh' here as the semantical primitive, not because it is the "best" primitive in some intuitive sense, but because rules concerning it are readily available. These rules, as we shall note in the next chapter, provide directly for a Boolean algebra based on inclusion. The character of non-translational semantics as a Boolean algebra of inclusion constructed upon the basis of elementary syntax is thus clearly brought to light. That semantics can be regarded merely as such an extended algebra has perhaps not been recognized heretofore.

From *GRC1–GRC14* we should very likely be able to prove any axioms we might wish to adopt characterizing any one of these alternative primitives. And conversely, if such axioms are to provide fully for the theory at hand. Therefore these various alternative ways of formulating non-translational semantics are presumably equivalent to each other.

We can also take certain *one-place* semantical predicates as primitive. First suppose a non-translational 'Tr' is taken as sole semantical primitive and characterized by appropriate axioms. All such axioms would presumably be provable within SM^{L}_{C}. But 'Cmprh' may be defined in terms of 'Tr' as follows.

'*a* Cmprh *b*' abbreviates '(PredCon *a* . PredCon *b* . Tr (*ex qu* (*b*⌢*ex hrsh a*⌢*ex*)))'.[1]

Hence upon suitably chosen axioms for 'Tr' *GRC1–GRC14* would themselves presumably be forthcoming as theorems within this meta-language. Thus the two meta-languages here also would be equivalent.

Another alternative is to take 'Univ' as a primitive in place of 'Tr'. The two are clearly interdefinable. Definitions of 'Tr' in terms of 'Univ' have been given above. But also

'Univ *a*' can abbreviate '(E*b*)(Tr *b* . *a* R^{invep}_{rp} *b* . AtUnivSent *b*)'.

Hence, on the basis of suitable axioms for 'Univ' taken as a primitive, the meta-language based on 'Tr' could be shown equivalent to that based on 'Univ'.

Another alternative is to be gained by taking 'Null' as sole semantical primitive. Then

[1] The author is indebted to Professor Alfred Tarski for this especially simple definition.

'Univ a' can abbreviate '(Eb)(Ec)(Ed)(Null b . Vbl c . b = (c⌢*invep*⌢d) . a = (c⌢*invep*⌢*tilde* d))'.

On the basis of suitable axioms the semantical meta-language based on 'Null' could also be shown equivalent with the other non-translational meta-languages.

Some of these various formulations of non-translational semantics are perhaps simpler and therefore preferable to others. But all the formulations are presumably equivalent and provide a minimal semantics for L, as we shall see in (XIII,F).

The discussion of non-translational semantics is continued in the next chapter.

CHAPTER IX

NON-TRANSLATIONAL SEMANTICS (II)

THE first sections of this chapter are devoted to drawing out the main consequences of the General Rules of Comprehension. In this way we shall gain a clearer picture than heretofore of the precise nature of SM_C^L. The theorems here are general in the sense that they hold for any object-language L of the kind being considered. Analogues of some of these theorems have already been cited in (IV,E and F). There they play a relatively minor role. Within SM_C^L, however, these theorems play an important role, helping to prove the basic theorems concerning the truth-concept. Within non-translational semantics, we do not allow the deductive devices of the object-language. The theorems of the general theory of comprehension in part compensate for this, and are a necessary prelude to the theory of truth (for a specific object-language).

The later sections of the chapter are concerned with the specific semantical properties of a particular object-language. As object-language we choose the Zermelo–Skolem system S discussed in Chapter VI. S is one of the most important and powerful first-order languages, and hence it will be of especial interest to study its semantical properties within a very narrow, non-translational, semantical meta-language. We can presuppose non-translational semantics formulated for S as object-language. We shall refer to this semantical meta-language as SM_C^S.

Most of the theorems mentioned or proved in §§ A–B are simple theorems of a *Boolean algebra* of predicate constants in the sense of (IV,F). Many of these theorems will be useful in the later sections of the chapter and in the sequel. The notion of the *semantical implexion* (or implication) of predicate constants is introduced in § C, and important semantical *theorems concerning truth* are proved in § D. To provide for the specific semantical properties of S, certain *additional assumptions* are required. These will be discussed in § E. In § F, it will be proved that, on suitable hypotheses, every theorem of S which is also a sentence is true in the appropriate sense. This theorem, as we know already, is essential for the proof of the semantical consistency of S. Finally, § G is devoted to a proof of the *consistency* of non-translational semantics relative to the consistency of the underlying logic and syntax.

A. Boolean Algebra. It has already been remarked that the general theory of comprehension consists essentially of elementary syntax together with a Boolean algebra of predicate constants. To verify this, we need merely note that the various theorems of a Boolean algebra are provable in SM^L_G. In particular we may prove analogues of the theorems of (IV,F) very briefly as follows.

Let $\boldsymbol{F}$ be any (one-place) abstract of SM^L_G containing primitively no occurrences of '$\frown$' or '$=$' or of any of the structural-descriptive constants. $\boldsymbol{F}$ then will contain primitively only expressional variables, parentheses, truth-functional symbols, '$\supset$', and 'Cmprh'. Let such an $\boldsymbol{F}$ be called a *purely semantical* abstract. A general law for co-extensiveness is then as follows.

TA1. $\vdash a$ CoExt $b :\supset: \boldsymbol{F}a .\equiv. \boldsymbol{F}b$, if $\boldsymbol{F}$ is a purely semantical abstract.

The proof is by induction (in the meta-language of SM^L_G).

Another co-extensiveness law we gain from *GRC13*.

TA2. $\vdash$ SentFuncOne a,b . c SF^d_b a . b FV $a :\supset: (b \frown invep \frown a)$ CoExt $(d \frown invep \frown c)$.

By *GRC13* we have that

$\vdash$ SentFuncOne a,b . c SF^d_b a . b FV a . $\sim d = b$. SentFuncOne c,d . a SF^b_d c . d FV $c :\supset: (b \frown invep \frown a)$ CoExt $(d \frown invep \frown c)$.

But by *TG8j*(III), we get *TA2* on condition that $\sim b = d$. But *TA2* on the condition that $b = d$ is immediate, using *TG8l*(III). Hence the theorem.

Also we have *TA2* without the hypothesis 'b FV a'.

TA3. $\vdash$ SentFuncOne a,b . c SF^d_b $a :\supset: (b \frown invep \frown a)$ CoExt $(d \frown invep \frown c)$.

The proof is by *TA2* and *GRC11*.

Clearly we can prove the requisite laws concerning semantical sums, in particular, the existence and uniqueness (to within co-extensiveness) law *TF4*(IV). The proof is by *GRC3*, *TA3*, and *TA1*. And similarly for products, the law *TF7*(IV).

Likewise we have the following laws concerning semantical negation.

TA4. $\vdash$ PredCon $a .\supset.$ (Eb)(PredCon b . b Neg a).

By *GRC5*.

TA5. $\vdash b$ Neg a . c Prod $a,b :\supset:$ Null c.

The law of reflexiveness for Sub, *TF1*(IV), may be proved here by *GRC6*, *TF7*(IV), *TA4*, and *TA5*. *TF2*(IV), the transitivity law for Sub, is immediate by *GRC2*. Also *TF3*, *TF5*, *TF6*, *TF8*, and *TF9* of Chapter IV are immediate in view of the definitions.

By *GRC8* and syntax we know that

TA6. $\vdash$ (E*a*)Univ *a*.

Also

TA7. $\vdash$ Univ *a* . *a* Sub *b* :$\supset$: Univ *b*,

and hence

TA8. $\vdash$ Univ *a* . Univ *b* :$\supset$: *a* CoExt *b*.

Combining *TA6* and *TA8* we get *TF12*(IV), the existence and uniqueness law for universal predicate constants.

Concerning semantically null predicate constants we have the following laws.

TA9. $\vdash$ *a* Neg *b* . Univ *b* :$\supset$: Null *a*.

Hence, using *TA4* and *TA6*, we have that

TA10. $\vdash$ (E*b*)Null *b*.

Also

TA11. $\vdash$ Null *a* . *b* Sub *a* :$\supset$: Null *b*,

and hence

TA12. $\vdash$ Null *a* . Null *b* :$\supset$: *a* CoExt *b*.

Thus we gain the existence and uniqueness law for semantically null predicate constants, *TF13*(IV).

Note also that

TA13. $\vdash$ *a* Sum *b*,*c* .v. *a* Prod *b*,*c* .v. (*a* Neg *b* . PredCon *c*) .v. ((Null *a* .v. Univ *a*) . PredCon *b* . PredCon *c*) :$\supset$: PredCon *a* . PredCon *b* . PredCon *c*.

TF15 and *TF16* of Chapter IV are immediate, using the definitions and *TA5*.

We may establish the distributive laws, *TF10* and *TF11* of Chapter IV, using the following lemmas.

TA14. $\vdash$ *c* Prod *a*,*b* :$\supset$: Null *c* .$\equiv$. (*d*)(*d* Sub *a* . *d* Sub *b* :$\supset$: Null *d*).

By *TA11*.

TA15. $\vdash$ PredCon *a* . PredCon *b* . $\sim$*a* Sub *b* :$\supset$: (E*c*)(*c* Sub *a* . $\sim$*c* Sub *b* . (*d*)(*d* Sub *c* . *d* Sub *b* :$\supset$: Null *d*)).

The proof utilizes *GRC6*. One of the main purposes of *GRC6* in fact is to enable us to prove this lemma. Its other purpose, as we have already noted, is to enable us to prove the reflexiveness law for Sub.

TA16. $\vdash$ PredCon *a* . PredCon *b* . $\sim$*a* Sub *b* :$\supset$: (E*c*)(E*d*)($\sim$Null *c* . *c* Sub *a* . *d* Prod *c*,*b* . Null *d*).

By *TA14* and *TA15*.

TA17. $\vdash d$ Sum a,b . c Sub d . e Prod c,a . e' Prod c,b . Null e . Null e' $:\supset:$ Null c.

The proof is by *GRC7* and *TA14*.

Using *TA16*, *TA17*, and *TA1* we can establish the distributive law *TF10*(IV). The other distributive law, *TF11*(IV), then also follows.

Finally, we may establish the following laws, which lead up to the uniqueness law for Neg.

TA18. $\vdash a$ Neg b . c Neg b . d Sum a,b $:\supset:$ d Sum c,b.

TA19. $\vdash d$ Sum a,b . d Sum c,b . e Prod a,b . Null e . e Prod c,b $:\supset:$ a CoExt e.

The proof of this is a little laborious.

We may now prove *TF14*(IV), the existence and uniqueness law for Neg, using also *TA4*.

B. Some Further Theorems. We note now a few further theorems. Some of these are simple Boolean laws which will be needed subsequently.

Because of its importance for a proof later the following may be stated as a theorem.

TB1. $\vdash$ SentFuncOne b',d . $b = (d \frown invep \frown b')$. SentFuncOne c',e . $c = (e \frown invep \frown c')$. d' SF^d_e c' . $a = (d \frown invep \frown (b' \text{ vee } d'))$ $:\supset:$ a Sum b,c.

The proof is by *TG9a*(III), *GRC3*, *TA3*, and *TA1*. An analogous law holds for products.

Given any predicate constants b and c, we have no right as yet to speak of *the* sum or product of b and c. We have proved merely that *at least one* sum or product exists. But in fact there is a multitude of sums or products. We thus have that

$\vdash$ PredCon a . PredCon b $:\supset:$ (Ec)(Ed)($\sim c = d$. c Sum a,b . d Sum a,b).

And similarly for products. In fact, given any finite number n, we can prove that at least n distinct sums (or products) of a and b exist. The reason is that we can always form a new sum (or product) simply by altering the abstract. E.g., if c is '$x \ni$ --- x ---' where c Sum a,b, we can let d be '$x \ni$(--- x --- . $x = x$)'. Evidently $\sim d = c$, yet both are sums of a and b. And similarly for products. The proofs of these statements, requiring some laws to be proved below, will be given in § D. (See *TD25*.)

We now list a few additional, useful Boolean laws.

TB2. $\vdash$ Null b . c Sum a,b $:\supset:$ c CoExt a,

TB3. $\vdash$ Null b . c Prod a,b $:\supset:$ c CoExt b,

TB4. $\vdash a$ Neg b .⊃. b Neg a,
TB5. $\vdash a$ Neg b . Null b :⊃: Univ a,
TB6. $\vdash$ Univ a . Null b :⊃: b Sub a,
TB7. $\vdash$ Univ a . Null b :⊃: ~a CoExt b,
TB8. $\vdash$ (Ea)(Eb)(~a CoExt b . PredCon a . PredCon b),

and

TB9. $\vdash$ Null a .⊃. ~Univ a.

It might be thought desirable to develop a more perspicuous notation for semantical sums, products, and negatives. Rather than to ascribe some property to a on the *hypothesis* that a is a semantical sum of b and c, we might wish to introduce a certain notation, say '$(b+c)$', for such a semantical sum, and then ascribe the property directly to this sum. This may be done as follows, using the notion of a selective description introduced in (II,F).

Let us define

'$(a+b)$' as '($\varepsilon c.c$ Sum a,b)',
'$(a.b)$' as '($\varepsilon c.c$ Prod a,b)',

and

'$-a$' as '($\varepsilon c.c$ Neg a)'.

'$(a+b)$' may be read 'any selected sum of a and b'. And similarly for the others. We can then prove the laws concerning sums, etc., by presupposing the laws already noted above. E.g., let us show that

$\vdash$ PredCon a . PredCon b :⊃: $(a+b)[(b+a)[(a+b)$ CoExt $(b+a)]]$.

Using the definition in (II,F), this can be written as follows.

$\vdash$ PredCon a . PredCon b :⊃: (Ec)c Sum a,b . (c)(c Sum a,b :⊃: (Ed) d Sum b,a . (d)(d Sum b,a .⊃. c CoExt d)).

This is provable using the associative and existence laws for semantical sums.

Although by using selective descriptions in this way we gain in perspicuity of notation, the proofs become longer and presuppose the theorems already noted. Hence little would be gained by using this notation throughout. The definitions here, however, are of interest in illustrating a usage for the Hilbert 'ε', which will also figure in an important way in some developments below in (XII,F).

C. Semantical Implexion of Predicate Constants. Because of the important role which the material conditional plays within the object-language, the corresponding semantical notion of the *implexion* (or *implication*) *of predicate constants* deserves special mention.

We say that an expression a is a *semantical implex*[1] of b with c, if and only if there is an expression d such that d is a semantical negative of b and a is a semantical sum of d and c.

'a Imp b,c' abbreviates '(Ed)(d Neg b . a Sum d,c)'.

This definition is analogous to *D1*, in (II,A), of the material conditional in terms of disjunction and negation. The notion of the semantical implexion of predicate constants will be especially useful in connection with the truth-concept as applied to sentences which are of the form of material conditionals.

Concerning semantical implexion we have first a law concerning its relata.

TC1. $\vdash a$ Imp b,c :⊃: PredCon a . PredCon b . PredCon c.

This follows from laws concerning Sum and Neg.

Just as in the case of Sum, Prod, and Neg, we have existence and uniqueness laws for Imp.

TC2. $\vdash$ PredCon b . PredCon c :⊃: (Ea)a Imp b,c.

TC3. $\vdash a$ Imp b,c . d Imp b,c :⊃: a CoExt d.

The proofs are by theorems concerning semantical negatives and sums.

Possibly the most useful law concerning the implexion of predicate constants is that where a is an implex of b with c and both a and b are universal then c is universal also. This law has some analogy with the *Modus Ponens* rule concerning the material conditional. In fact, it will be useful in dealing with the latter, as we shall note below (*TD7*).

TC4. $\vdash a$ Imp b,c . Univ a . Univ b :⊃: Univ c.

The proof is by some Boolean laws.

Other useful laws concerning semantical implexion are the following.

TC5. $\vdash$ SentFuncOne c,b . SentFuncOne d,b . $a = (b \frown invep \frown (c\ hrsh\ d))$. $c' = (b \frown invep \frown c)$. $d' = (b \frown invep \frown d)$:⊃: a Imp c',d'.

TC6. $\vdash$ (Eb')(Ec')(Ed)(Ed')(Ee)(SentFuncOne b',d . SentFuncOne c',e . $b = (d \frown invep \frown b')$. $c = (e \frown invep \frown c')$. $d'\ \mathrm{SF}^d_e\ c'$. $a = (d \frown invep \frown (b'\ hrsh\ d'))$) .⊃. a Imp b,c.

TC7. $\vdash$ SentFuncOne b,a . SentFuncOne c,a :⊃: Univ $(a \frown invep \frown (b\ hrsh\ c))$.≡. $(a \frown invep \frown c)$ Cmprh $(a \frown invep \frown b)$.

We go on now to some theorems in the theory of truth which in no

[1] The word 'implex' is due to Quine. See his *A System of Logistic* (Harvard University Press, Cambridge: 1934), p. 91. Quine's usage, however, differs from the present semantical one.

way depend upon the non-logical rules of L. Many of these theorems have already been proved in SM_D^L above. There they depend upon the Adequacy Rule. Here in SM_G^L, which lacks an Adequacy Rule, they must be proved from *GRC1–GRC14* in an altogether new way. Because of the importance of these theorems in non-translational semantics, the proofs will be given in some detail.

D. Logical Theorems and Truth. An immediate consequence of *GRC8* is to the effect that all logical axioms of L (which are also sentences) and closures of such are true in the sense of the definition in terms of vacuous abstracts ((V,B) and (VIII,C)). We can then show also that all logical theorems of L which are sentences are true. In this section we prove in SM_G^L these two basic theorems and some other important theorems concerning truth.

TD1. $\vdash$ LogAx a :$\supset$: (Eb)(b Clsr′ a . Tr b).

The proof is by *GRC8*.

As a consequence of *GRC12* we have that

TD2a. $\vdash$ Sent a . Vbl b :$\supset$: Univ ($b \frown invep \frown a$) .v. Univ ($b \frown invep \frown tilde\ a$),

using *TB5* and *GRC5*.

Using *TD2a*, we can prove immediately that given a sentence a either it is true or its negative is.

TD2b. $\vdash$ Sent a :$\supset$: Tr a .v. Tr (*tilde a*).

For we have first that, by the definition of 'Tr',

$\vdash$ Sent a . Vbl b . $\sim$Tr a :$\supset$: $\sim$Univ ($b \frown invep \frown a$).

Hence by *TD2a*,

$\vdash$ Sent a . Vbl b . $\sim$Tr a :$\supset$: Vbl b . Univ ($b \frown invep \frown tilde \frown a$).

Hence the theorem, using *TG8e*(III).

The following theorem gives an important property of vacuous abstraction.

TD3. $\vdash$ Sent a . Vbl b . Vbl c . $\sim b = c$. Univ ($b \frown invep \frown a$) :$\supset$: Univ ($c \frown invep \frown a$).

The proof is immediate using *GRC11* and *TA7*.

The next theorem states that any sentence a is false (not true) if and only if (*tilde a*) is true.

TD4. $\vdash$ Sent a :$\supset$: $\sim$Tr a .$\equiv$. Tr (*tilde a*).

It is an immediate consequence of *TD2b* that

(1) $\vdash$ Sent a :$\supset$: $\sim$Tr a .$\supset$. Tr (*tilde a*).

Hence we need prove merely that

(2) $\vdash$ Sent a :⊃: Tr (*tilde a*) .⊃. ~Tr a.

But this may be established if we note that, by *GRC5* and *TB4*,

$\vdash$ Sent a . Vbl b . Univ ($b \frown$ *invep* $\frown$ *tilde* a) . Sent (*tilde* a) :⊃: ($b \frown$ *invep* $\frown a$) Neg ($b \frown$ *invep* $\frown$ *tilde* a).

And hence, using *TA9* and *TB9*,

$\vdash$ Sent a . Vbl b . Univ ($b \frown$ *invep* $\frown$ *tilde* $\frown a$) . Sent (*tilde* a) :⊃: Sent a . Vbl b. ~Univ ($b \frown$ *invep* $\frown a$).

Using now *TD3*, we have that

$\vdash$ Sent a . Tr (*tilde* a) :⊃: Sent a . (c)(Vbl c .⊃. ~Univ ($c \frown$ *invep* $\frown a$)).

From this, we get (2) and hence the theorem.

Also, we note as another consequence of *GRC12*, or rather of *TD2a*, that

TD5a. $\vdash$ Sent a . Vbl b . ~Univ ($b \frown$ *invep* $\frown a$) :⊃: Null ($b \frown$ *invep* $\frown a$).

By *TD2a*, *GRC5*, and *TB4*,

$\vdash$ Hyps :⊃: Univ ($b \frown$ *invep* $\frown$ *tilde* a) . ($b \frown$ *invep* $\frown a$) Neg ($b \frown$ *invep* $\frown$ *tilde* a).

('Hyps' here stands for 'hypotheses'. The usage intended here and throughout will be clear from the context.) Then by *TA9* we immediately get *TD5a*.

Also from *TD5a* we can prove that

TD5b. $\vdash$ Sent a . Vbl b . ~Tr a :⊃: Null ($b \frown$ *invep* $\frown a$).

Another important theorem concerning the truth-concept is the principle that if a disjunction (*a vee b*) of sentences a and b is true then at least one of the disjuncts is true.

TD6. $\vdash$ Sent a . Sent b . Tr (*a vee b*) :⊃: Tr a .v. Tr b.

First we note that, by *GRC3*,

(1) $\vdash$ Vbl c . Sent a . Sent b . Univ ($c \frown$ *invep* $\frown$ (*a vee b*)) . Sent (*a vee b*) :⊃: ($c \frown$ *invep* $\frown$ (*a vee b*)) Sum ($c \frown$ *invep* $\frown a$),($c \frown$ *invep* $\frown b$) . Univ ($c \frown$ *invep* $\frown$ (*a vee b*)).

Suppose that ~Tr a and ~Tr b. We can then show, using *TD5b*, that

(2) $\vdash$ Hyps (of (1)) . ~Tr a . ~Tr b :⊃: ($c \frown$ *invep* $\frown$ (*a vee b*)) Sum ($c \frown$ *invep* $\frown a$), ($c \frown$ *invep* $\frown b$) . Null ($c \frown$ *invep* $\frown a$) . Null ($c \frown$ *invep* $\frown b$) . Univ ($c \frown$ *invep* $\frown$ (*a vee b*)).

But by *TB2* and *TA1* we would then have that

$\vdash$ Hyps (of (2)) . Concls (of (2)) :⊃: Null ($c \frown$ *invep* $\frown$ (*a vee b*)) . Univ ($c \frown$ *invep* $\frown$ (*a vee b*)).

('Concls' here stands for 'conclusions' in an obvious way.) This is impossible by *TB9*, and hence the theorem.

An immediate consequence of *TD6* and *TD4* is the following, the *Modus Ponens* principle for the truth-concept.

TD7. $\vdash$ Tr *a* . Tr (*a hrsh b*) :⊃: Tr *b*.

By *TD6*, we have that

$\vdash$ Sent (*tilde a*) . Sent *b* . Tr (*tilde a vee b*) . ~Tr (*tilde a*) :⊃: Tr *b*.

Using the definition of '*hrsh*', *TD4*, *TG10a*(III), and *TG8e*(III), we get the theorem. Similarly we have the following principle which corresponds with the *Rule of Generalization*.

TD8. $\vdash$ Tr *a* . *b* Gen *a* :⊃: Tr *b*.

By *GRC10* we have that

$\vdash$ Sent *a* . Vbl *c* . Univ (*c*∩*invep*∩*a*) . Vbl *d* :⊃: Univ (*c*∩*invep*∩ *d qu a*),

and as a result that

$\vdash$ Sent *a* . Vbl *c* . Univ (*c*∩*invep*∩*a*) . Vbl *d* . *b* = (*d qu a*) :⊃: Sent *b* . Vbl *c* . Univ (*c*∩*invep*∩*b*).

Hence the theorem.

Combining *TD7* and *TD8*, we have that if *a* and *b* are true and *c* is an immediate consequence of them, then *c* is also true.

TD9. $\vdash$ Tr *a* . Tr *b* . *c* IC *a*,*b* :⊃: Tr *c*.

We are now in a position to prove the second main theorem of this section, that if *a* is a sentence and is a logical theorem of *L*, then it is also true in the appropriate sense.

TD10. $\vdash$ Sent *a* . LogThm *a* :⊃: Tr *a*.

The proof utilizes the Rule of Framed Ingredients, *TG4*(III), and is similar to that of *TE5*(V).

Using *TD10* and *TD7* we can now establish what is essentially the converse of the theorem stipulated by *TD6*.

TD11. $\vdash$ (Tr *a* . Sent *b*) .v. (Tr *b* . Sent *a*) :⊃: Tr (*a vee b*).

By *TD7*,

$\vdash$ Tr *a* . Tr (*a hrsh* (*a vee b*)) :⊃: Tr (*a vee b*).

But by *TD10*,

$\vdash$ Sent *a* . Sent *b* :⊃: Tr (*a hrsh* (*a vee b*)),

noting that

$\vdash$ Sent *a* . Sent *b* :⊃: LogThm (*a hrsh* (*a vee b*)).

Hence

$$\vdash \text{Sent } b \,.\, \text{Tr } a :\supset: \text{Tr } (a \textit{ vee } b).$$

In a similar way we can show that

$$\vdash \text{Sent } a \,.\, \text{Tr } b :\supset: \text{Tr } (a \textit{ vee } b).$$

Combining these we get *TD11*.

Combining *TD6* and *TD11* we have that

TD12. $\vdash$ Sent a . Sent b :$\supset$: (Tr a .v. Tr b) .$\equiv$. Tr (a *vee* b).

From *TD12* using *TD4* we can prove immediately that

TD13. $\vdash$ Sent a . Sent b :$\supset$: (Tr a .$\supset$. Tr b) .$\equiv$. Tr (a *hrsh* b).

Hence also,

TD14. $\vdash$ Sent a . Sent b :$\supset$: (Tr a . Tr b) .$\equiv$. Tr (a *dot* b),

and

TD15. $\vdash$ Sent a . Sent b :$\supset$: (Tr a .$\equiv$. Tr b) .$\equiv$. Tr (a *tripbar* b).

Also combining *TD10* and *TD7* we prove that

TD16. $\vdash$ Tr a . LogThm (a *hrsh* b) . Sent b :$\supset$: Tr b.

The following theorem is especially interesting in showing that an atomic universal sentence '(x)---x---' of L is true if and only if the corresponding abstract '$x\mathfrak{z}$---x---' is universal. From this one can immediately establish that truth as defined in terms of vacuous abstracts coincides with truth defined in one of the other ways mentioned in (VIII,C), if applied to atomic universal sentences.

TD17. $\vdash$ Sent (a *qu* b) . Vbl a :$\supset$: Tr (a *qu* b) .$\equiv$. Univ ($a \frown$ *invep* $\frown b$).

This we prove in two parts, first that

(1) $\vdash$ Sent (a *qu* b) . Vbl a . Tr (a *qu* b) :$\supset$: Univ ($a \frown$ *invep* $\frown b$),

and then that

(2) $\vdash$ Sent (a *qu* b) . Vbl a . Univ ($a \frown$ *invep* $\frown b$) : $\supset$: Tr (a *qu* b).

For the proof of (1), we note that, by *TC5*,

(3) $\vdash$ Sent (a *qu* b) . Vbl a :$\supset$: ($a \frown$ *invep* $\frown$ (a *qu* b *hrsh* b)) Imp ($a \frown$ *invep* $\frown a$ *qu* b),($a \frown$ *invep* $\frown b$).

Also, using *TD3* and the definition of 'Tr', we know that

(4) $\vdash$ Hyps (of (1)) :$\supset$: Univ ($a \frown$ *invep* $\frown a$ *qu* b).

But also, because from syntax

$$\vdash \text{Sent } (a \textit{ qu } b) \,.\, \text{Vbl } a :\supset: \text{LogAx } (a \textit{ qu } b \textit{ hrsh } b),$$

we know using *GRC8* that

(5) $\vdash$ Hyps (of (1)) :$\supset$: Univ ($a \frown$ *invep* $\frown$ (a *qu* b *hrsh* b)).

Combining now (3), (4), and (5), and using *TC4*, we get (1). Now for (2). Using *GRC10* and the definition of 'Tr', we have immediately that

(6) $\vdash$ Sent $(a\ qu\ b)$. Vbl a . Univ $(a \frown invep \frown b)$. a FV b :$\supset$: Tr $(a\ qu\ b)$.

But also

(7) $\vdash$ Sent $(a\ qu\ b)$. Vbl a . Univ $(a \frown invep \frown b)$. $\sim a$ FV b :$\supset$: Tr $(b\ hrsh\ a\ qu\ b)$,

by *TD10*. Also

(8) $\vdash$ Sent $(a\ qu\ b)$. Vbl a . Univ $(a \frown invep \frown b)$. $\sim a$ FV b :$\supset$: Tr b,

because,

$\vdash$ Univ $(a \frown invep \frown b)$.$\supset$. PredCon $(a \frown invep \frown b)$,

by *TA13*, and by *TG8k* and *TG8b* of III,

$\vdash$ PredCon $(a \frown invep \frown b)$. Vbl a . $\sim a$ FV b :$\supset$: Sent b.

Combining (7) and (8) and using *TD7* we have that

$\vdash$ Sent $(a\ qu\ b)$. Vbl a . Univ $(a \frown invep \frown b)$. $\sim a$ FV b :$\supset$: Tr $(a\ qu\ b)$.

This together with (6) gives (2), so that 'Tr $(a\ qu\ b)$' holds on the hypotheses of (2) whether a is a free variable of b or not. But (2) and (1) together give the theorem.

A consequence of *TD17* is that

TD18. $\vdash$ SentFuncOne b,a :$\supset$: Univ $(a \frown invep \frown b)$.$\equiv$. Univ $(a \frown invep \frown a\ qu\ b)$.

By *GRC10*

(1) $\vdash$ SentFuncOne b,a . Univ $(a \frown invep \frown b)$:$\supset$: Univ $(a \frown invep \frown a\ qu\ b)$.

Now conversely, by *TD17*,

$\vdash$ Sent $(a\ qu\ b)$. Vbl a . Vbl c . Univ $(c \frown invep \frown a\ qu\ b)$:$\supset$: Univ $(a \frown invep \frown b)$,

and hence

(2) $\vdash$ SentFuncOne b,a . Univ $(a \frown invep \frown a\ qu\ b)$:$\supset$: Univ $(a \frown invep \frown b)$.

Combining (1) and (2) one gets the theorem.

The next theorem interrelates the truth-concept as applied to sentences of the form of universal conditionals with subsumption.

TD19. $\vdash$ SentFuncOne a,c . SentFuncOne b,c . Tr $(c\ qu\ (a\ hrsh\ b))$:$\supset$: $(c \frown invep \frown a)$ Sub $(c \frown invep \frown b)$.

By *TD17*, we know that, on the hypothesis, Univ $(c \frown invep \frown (a\ hrsh$

b)). But using then *TC7* we get the theorem immediately. Similarly we can prove that

TD20. $\vdash$ SentFuncOne a,c . SentFuncOne b,c . Tr (c *qu* (a *tripbar* b)) $:\supset:$ ($c \frown$ *invep* $\frown a$) CoExt ($c \frown$ *invep* $\frown b$).

The converses of the theorems stipulated by *TD19* and *TD20* also hold. Hence

TD21. $\vdash$ SentFuncOne a,c . SentFuncOne b,c $:\supset:$ Tr (c *qu* (a *hrsh* b)) $.\equiv.$ ($c \frown$ *invep* $\frown a$) Sub ($c \frown$ *invep* $\frown b$),

and

TD22. $\vdash$ SentFuncOne a,c . SentFuncOne b,c $:\supset:$ Tr (c *qu* (a *tripbar* b)) $.\equiv.$ ($c \frown$ *invep* $\frown a$) CoExt ($c \frown$ *invep* $\frown b$).[1]

Also we have the following co-extensiveness law.

TD23. $\vdash$ PredCon a . Vbl b $:\supset:$ a CoExt ($b \frown$ *invep* $\frown a \frown b$).

By *GRC14*, *TA1*, and *GRC13* we know that

$$\vdash \text{Vbl } b :\supset: \textit{pee} \text{ CoExt } (b \frown \textit{invep} \frown \textit{pee} \frown b).$$

By *TD20* and *TD10* (recalling *Abst*), we have that

$\vdash$ SentFuncOne a,b . c = ($b \frown$ *invep* $\frown a$) $:\supset:$ c CoExt ($b \frown$ *invep* $\frown c \frown b$).

Using *GRC13* again and combining these two, we get the theorem.

Using *TD23*, *TA1*, and *TD19* we can also prove that

TD24. $\vdash$ PredCon a . PredCon b . Vbl c $:\supset:$ b Cmprh a $.\equiv.$ Tr (c *qu* ($a \frown c$ *hrsh* $b \frown c$)).

Also now that we have *TD20* we can prove the law that, given any two predicate constants a and b at least two semantical sums of a and b exist.

TD25. $\vdash$ PredCon a . PredCon b $:\supset:$ (Ec)(Ed)($\sim c = d$. c Sum a,b . d Sum a,b).

[1] *TD21* is especially interesting in giving a property of the truth-concept as applied to sentences of the universal conditional form. It is interesting to compare this theorem with a famous passage from Hobbes, with whose doctrine it seems to be in essential agreement. "When two Names are joyned together into a Consequence, or Affirmation; as thus, *A man is a living creature*; or thus, *if he be a man, he is a living creature*, If the latter name *Living creature* signifie all that the former name *Man* signifieth, then the affirmation, or consequence is *true*; otherwise *false*. For *True* and *False* are attributes of Speech, not of Things. . . . Seeing then that *truth* consisteth in the right ordering of names in our affirmations, a man that seeketh precise *truth*, had need to remember what every name he uses stands for . . ." (Hobbes, *op. cit.*, p. 15). According to *TD21*, a universal conditional is true if and only if the "latter name" comprehends the former.

Using *TB1* above we have first that

(1) $\vdash$ SentFuncOne b',d . $b = (d \frown invep \frown b')$. SentFuncOne c',e . $c = (e \frown invep \frown c')$. d' SF^d_e c' . $a = (d \frown invep \frown (b'$ *vee* $d'))$:$\supset$: a Sum b,c.

But also using *TD10*,

(2) $\vdash$ SentFuncOne b',d . SentFuncOne d',d :$\supset$: Tr (d *qu* (b' *vee* d') *tripbar* d *qu* ((b' *vee* d') *dot* (b' *vee tilde* b'))).

Hence using *TD20*,

$\vdash$ Hyps (of (2)) :$\supset$: ($d \frown invep \frown (b'$ *vee* $d'))$ CoExt ($d \frown invep \frown ((b'$ *vee* d') *dot* (b' *vee tilde* b'))).

Then, using (1) and some syntactical theorems,

$\vdash$ Hyps (of (1)) . $a' = (d \frown invep \frown ((b'$ *vee* $d)$ *dot* (b' *vee tilde* $b'))) :\supset:$ a Sum b,c . a' CoExt a . $\sim a' = a$.

From this we get the theorem, using *TA1* and eliminating the unneeded hypotheses.

In a similar way we can prove the corresponding laws concerning semantical products and negatives, and concerning universal and null predicate constants. Also these laws may obviously be generalized to provide the existence of any pre-assigned number n of semantical sums, products, etc.

E. Specific Hypotheses of Comprehension. Let us now take L as S, as in Chapter VI, and study some of its specific semantical properties, i.e., semantical properties which depend upon the non-logical axioms of S. The meta-language here is SM^S_C.

The general semantical framework for S is provided by the material of the preceding sections. Especially important among the general semantical theorems concerning L (taken now as S) given is the theorem *TD10* to the effect that all logical theorems of S which are also sentences are true. An important semantical theorem of the present section will be to the effect that all theorems of S, logical or *non-logical*, which are sentences are also true in the appropriate sense. To prove this in SM^S_C, certain additional assumptions would be required. Corresponding to each non-logical axiom of S we should need here a rule stating that a certain abstract of S has such and such a semantical property. If the axiom of S is a sentence a, the corresponding rule of SM^S_C would state in effect that an abstract ($b \frown invep \frown a$), where b is a variable, is universal. Some such rules as these would seem essential to enable us to prove first that all non-logical axioms of S (which are sentences) and closures of such are true, and then that all theorems whatsoever of S which are sentences are true. None of the General Rules of Comprehension depends upon the

axiomatic structure of the object-language, other than that the object-language contain all theorems of a simple, applied, logical calculus of first order with identity. But the non-logical part, so to speak, of the object-language is in no way characterized by the General Rules of Comprehension.

The task of non-translational semantics, as conceived here, is merely to provide the general framework for a semantical definition of truth. The rules of the non-translational meta-language should contain nothing dubious and should be quite independent of any of the specific, non-logical assumptions of a particular object-language. The general theory of comprehension is in accord with this conception. It provides the general framework for a semantical definition of truth, but does not enable us to determine which sentences of the object-language are true (other than those sentences which are logical theorems). Also the assumptions *GRC1–GRC14* contain nothing dubious, in the sense roughly that if elementary syntax is consistent then the general theory of comprehension is also, as we shall see in § G. And none of the assumptions of SM_C^S in any way depends upon the non-logical axioms of the object-language.

Hence we shall not wish to add any further axioms to SM_C^S, and we shall not be able to prove within SM_C^S that all the theorems of S which are sentences are true. But we can formulate within SM_C^S sentences stating that certain predicate constants of S are universal or non-null. Statements of this kind can be taken as hypotheses wherever needed. In this way we can prove, on certain hypotheses, what could be proved otherwise by assuming rules stating that those axioms of S which are sentences are true. (This device of using such hypotheses where needed is essentially that of *Principia Mathematica* with regard to the Axiom of Infinity and the Multiplicative Axiom.)[1]

The hypotheses required for S are called *Specific Hypotheses of Comprehension*, and will correspond with the non-logical or set-theoretical rules of S. For each such rule there will be a corresponding Specific Hypothesis. These hypotheses may be listed as follows.

Corresponding to the rule *S1* of Extensionality (VI,A and B), we define here within SM_C^S,

$$\text{'}SHC1\text{'} \text{ as } \text{'}(a)(b)(\text{ExtAx } a \,.\, \text{Vbl } b :\supset: \text{Univ } (b \frown invep \frown a))\text{'}.$$

'*SHC1*' is an abbreviation in SM_C^S for a statement that for any expression a which is an Axiom of Extensionality and for any variable b, the vacuous abstract $(b \frown invep \frown a)$ is universal. E.g., according to

[1] Cf. *Principia Mathematica*, vol. 1, pp. 536–42 and *passim*, and vol. 2, pp. 200–25 and *passim*.

this hypothesis

$$`x \ni (x)(x')((x'')(x'' \varepsilon x .\equiv. x'' \varepsilon x') .\supset. x = x')'$$

is universal.

Corresponding to the rule *S2* of *S*, providing for the Existence of the Null Set, we let

'SHC2' abbreviate '(*a*)(*b*)(NullSetAx *a* . Vbl *b* :⊃: Univ (*b*∩*invep* *a*))'.

According to this hypothesis,

$$`x''' \ni (Ex)(x') \sim x' \varepsilon x',$$

e.g., is universal.

In a similar way we can define Specific Hypotheses of Comprehension *SHC3*–*SHC7* corresponding to the rules *S3*–*S7* of *S*. Corresponding to *S8*, *S9*, and *S10* we define Specific Hypotheses of Comprehension as follows.

'SHC8' abbreviates '(*a*)(*b*)(*c*)(SubsetFormAx *a* . *b* Clsr' *a*. Vbl *c* :⊃: Univ (*c*∩*invep*∩*b*))'.

And similarly for the others. Then we can let

'SHC' abbreviate '(*SHC1* . *SHC2*. ... . *SHC10*)'.

The theorems in § F below will be proved on the hypothesis that *'SHC'* holds. But whether *'SHC'* holds or not it is not the task of the non-translational semantics of *S* to decide.

Note that *'SHC'* may also be defined within any of the alternative formulations of non-translational semantics considered in (VIII,E). But no such definition would be suitable unless *'SHC'* as defined in one meta-language is equivalent with *'SHC'* as defined in any of the others. But such equivalent definitions can easily be given.

F. Set-Theoretical Theorems and Truth. We may now prove some laws on the hypothesis that *'SHC'* holds in SM_C^S. In particular we prove within SM_C^S that on this hypothesis all theorems of *S* which are sentences are true.

A few lemmas leading up to this theorem are as follows. The first lemma states in effect that, on the hypothesis *'SHC'*, if *a* is a set axiom of *S* or closure of such, then it is true.

TF1a. ⊢ *SHC* . (ExtAx *a* .v. NullSetAx *a* .v. PairAx *a* .v. SumAx *a* .v. PowerAx *a* .v. ChoiceAx *a* .v. InfAx *a* .v. (E*b*)((SubsetFormAx *b* .v. ReplAx *b* .v. FoundAx *b*) . *a* Clsr' *b*)) :⊃: Tr *a*.

The proof is immediate. Also

TF1b. ⊢ *SHC* . SetAx *a* :⊃: (E*b*)(*b* Clsr' *a* . Tr *b*).

We can now show that all *theorems* of S which are sentences are true on the hypothesis '*SHC*'. The method is similar to that used in showing that all *logical theorems* which are sentences are true in § D above.

TF2. $\vdash$ *SHC* . Sent *a* . Thm *a* :$\supset$: Tr *a*.

Having now the crucial theorem *TF2*, we are in a position to prove the semantical consistency of S on the hypothesis that '*SHC*' holds. The proof is similar to that of (V,E).

TF3. $\vdash$ *SHC* . Sent *a* :$\supset$: $\sim$(Thm *a* . Thm (*tilde a*)).

TF4. $\vdash$ *SHC* . Fmla *a* :$\supset$: $\sim$(Thm *a* . Thm (*tilde a*)).

The following theorem states that on the hypothesis '*SHC*' if *a* is true and (*a hrsh b*) is a theorem of S where *b* is a sentence, then *b* is also true.

TF5. $\vdash$ *SHC* . Tr *a* . Thm (*a hrsh b*) . Sent *b* :$\supset$: Tr *b*.

The proof is by *TF2* and *TD7*.

The following two theorems are immediate consequences of *TD19* and *TD20*, using *TF2*.

TF6. $\vdash$ *SHC* . SentFuncOne *a,c* . SentFuncOne *b,c* . Thm (*c qu* (*a hrsh b*)) :$\supset$: (*c*$\frown$*invep*$\frown$*b*) Cmprh (*c*$\frown$*invep*$\frown$*a*).

TF7. $\vdash$ *SHC* . SentFuncOne *a,c* . SentFuncOne *b,c* . Thm (*c qu* (*a tripbar b*)) :$\supset$: (*c*$\frown$*invep*$\frown$*a*) CoExt (*c*$\frown$*invep*$\frown$*b*).

These theorems are of interest in connecting comprehension and co-extensiveness with theorems of S of such and such a form.

G. Relative Consistency of Non-Translational Semantics. We now leave the discussion of the specific semantics of S and turn to a proof of the relative consistency of SM_C^L, for any L with '='.

In (V,G) the consistency of the semantical meta-language SM_D^L was proved relative to that of the language consisting jointly of M together with the translation of L. We have already noted that analogues of the axioms of SM_C^L are provable within SM_D^L, taking 'Cmprh', 'Sum', etc., to be defined within SM_D^L in terms of 'Den'. The logical and syntactical axioms of SM_C^L are precisely the corresponding axioms of SM_D^L, and the Rules of Inference of the two semantical meta-languages are similar. Hence, on the assumption that the joint language consisting of M and the translation of L is consistent, it follows that SM_C^L is consistent likewise.

But we can also show that SM_C^L is consistent simply on the supposition that M is. For this we seek an interpretation of 'Cmprh' within M, i.e., an interpretation of a *purely syntactical kind* such that *GRC1–GRC14* become theorems of M.

We let

'Thm′ a' abbreviate '(Eb)(a FrIng b . (c)(c FrIng b :⊃: LogAx c .v. c = ($ex \frown id \frown ex \frown ac$) .v. (E$d$)(E$e$)($d$ Pr_b c . e Pr_b c . c IC d,e)))'.

Clearly the notion Thm′ is very close to that of Thm for L^* (V,F). (Take 'P' of L^* as '$x\varepsilon x = x$'.) Hence many of the foregoing theorems concerning Thm, in particular those of (III,G), also hold of Thm′. Such theorems will be referred to here by referring to the corresponding theorems of (III,G).

Let 'b Cmprh′ a' be taken to express that b and a are predicate constants and that (i) where b is ($c \frown invep \frown d$), d being a sentential function of the one variable c, and a is ($e \frown invep \frown a'$), a' being a sentential function of the one variable e, the formula (a' *hrsh* d) is a Thm′ of L, or (ii) b and a are *pee* (and ($pee \frown ex$ *hrsh* $pee \frown ex$) is a Thm′ of L), or (iii) b is *pee* and a is ($e \frown invep \frown a'$), a' being a sentential function of the one variable e, and (a' *hrsh* $pee \frown ex$) is a Thm′ of L, or (iv) a is *pee* and b is ($c \frown invep \frown d$), d being a sentential function of the one variable c, and ($pee \frown ex$ *hrsh* d) is a Thm′ of L. For this purely syntactical notion we shall use 'Cmprh″' in place of 'Cmprh'. We let

'(a *abst* b)' abbreviate '($a \frown invep \frown b$)'.

'Cmprh″' is then definable within M as follows.

'b Cmprh′ a' abbreviates '((Ec)(Ed)(Ee)(Ea')(b = (c *abst* d) . SentFuncOne d,c . a = (e *abst* a') . SentFuncOne a',e . Thm′ (a' *hrsh* d)) .v. (a = *pee* . b = *pee* . Thm′ ($pee \frown ex$ *hrsh* $pee \frown ex$)) .v. (Ee)(Ea')(SentFuncOne a',e . a = (e *abst* a') . b = *pee* . Thm′ (a' *hrsh* $pee \frown ex$)) .v. (Ec)(Ed)(SentFuncOne d,c . b = (c *abst* d) . a = *pee* . Thm′ ($pee \frown ex$ *hrsh* d)))'.

(Note that if L contained primitive one-place predicate constants other than 'P', clauses would have to have added in the definiens here to provide for them. Also the clause 'Thm′ ($pee \frown ex$ *hrsh* $pee \frown ex$)' may be dropped.)

Let *GRC1′* differ from *GRC1* only in containing 'Cmprh″' wherever the primitive expansion of *GRC1* contains 'Cmprh'. In a similar way we form *GRC2′*, *GRC3′*, . . . , and *GRC14′*. We now show somewhat sketchily that *GRC1′*–*GRC14′* are provable in M. Some of the proofs are long but not essentially difficult.

GRC1′ is that

⊢ (Eb)(a Cmprh′ b .v. b Cmprh′ a) .≡. PredCon a.

But this can be proved immediately if we note that

⊢ SentFuncOne d,c . a = (c *abst* d) :⊃: (Ea')(Eb')(SentFuncOne a',b' . SentFuncOne d,c : Thm′ (a' *hrsh* d) .v. Thm′ (d *hrsh* a')) .≡. PredCon a,

and that

$\vdash$ (Ea')(Eb')(SentFuncOne a',b' : Thm′ (a' *hrsh* $pee \frown ex$) .v. Thm′ ($pee \frown ex$ *hrsh* a')) .≡. PredCon *pee*.

We then get *GRC1′* by combining these.

GRC2′ is also provable if we note that

$\vdash$ SentFuncOne a',d . a = (d *abst* a') . SentFuncOne b',e . b = (e *abst* b') . Thm′ (b' *hrsh* a') . SentFuncOne c',d' . c = (d' *abst* c') . SentFuncOne a'',e' . a = (e' *abst* a'') . Thm′ (a'' *hrsh* c') :⊃: a'' = a' . SentFuncOne c',d' . c = (d' *abst* c') . SentFuncOne b',e . b = (e *abst* b') . Thm′ (b' *hrsh* c').

Similarly

$\vdash$ SentFuncOne a',d . a = (d *abst* a') . b = *pee* . Thm′ ($pee \frown ex$ *hrsh* a') . SentFuncOne c',d' . c = (d' *abst* c') . SentFuncOne a'',e' . a = (e' *abst* a'') . Thm′ (a'' *hrsh* c') :⊃: a'' = a' . SentFuncOne c',d' . c = (d' *abst* c') . b = *pee* . Thm′ ($pee \frown ex \frown hrsh \frown c'$),

and so on, where in place of 'a', 'b', and 'c' in *GRC2′* we put in '*pee*' in all possible ways. Combining these we have *GRC2′* itself.

Before considering the analogue of *GRC3*, the Sum Rule, we should note that a notion of Sum as defined in (VIII,B) is now definable in terms of 'Cmprh''. This notion we call 'Sum''.

That *GRC3′*, which we shall call the *Sum′ Rule*, holds in *M* can be seen as follows. First we note that

(1) $\vdash$ SentFuncOne b',d . SentFuncOne c',d . b = (d *abst* b') . c = (d *abst* c') . a = (d *abst* (b' *vee* c')) :⊃: Thm′ (b' *hrsh* (b' *vee* c')) . Thm′ (c' *hrsh* (b' *vee* c')).

Also

$\vdash$ Hyps (of (1)) . SentFuncOne a'',d . Thm′ (b' *hrsh* a'') . Thm′ (c' *hrsh* a'') :⊃: Thm′ ((b' *vee* c') *hrsh* a'').

From this and (1) we have that

(2) $\vdash$ Hyps (of (1)) :⊃: (SentFuncOne b',d . SentFuncOne (b' *vee* c'),d . Thm′ (b' *hrsh* (b' *vee* c'))) . (SentFuncOne c',d. SentFuncOne (b' *vee* c'),d . Thm′ (c' *hrsh* (b' *vee* c'))) . ((SentFuncOne b',d . SentFuncOne a'',d . Thm′ (b' *hrsh* a'')) . (SentFuncOne c',d . SentFuncOne a'',d . Thm′ (c' *hrsh* a'')) :⊃: (SentFuncOne (b' *vee* c'),d . SentFuncOne a'',d . Thm′ ((b' *vee* c') *hrsh* a''))),

using the syntactical theorem (cf. *TG9a*(III)) that

$\vdash$ SentFuncOne b',d . SentFuncOne c',d :⊃: SentFuncOne (b' *vee* c'),d.

But from (2) we have that

$\vdash$ Hyps (of (1)) :⊃: *a* Cmprh′ *b* . *a* Cmprh′ *c* . (*a*″)(*d*′)(SentFuncOne *a*″,*d*′ . (*d*′ *abst a*″) Cmprh′ *b* . (*d*′ *abst a*″) Cmprh′ *c* :⊃: (*d*′ *abst a*″) Cmprh′ *a*),

and hence that

(3) $\vdash$ Hyps (of (1)) :⊃: (*d abst* (*b*′ *vee c*′)) Sum′ (*d abst b*′),(*d abst c*′),

or that

$\vdash$ Hyps (of (1)) :⊃: *a* Sum′ *b*,*c*.

Hence *GRC3*′. (The additional clauses concerning *pee* cannot hold on the hypotheses and hence may be added as disjuncts wherever needed.)

In a similar way one proves the analogue *GRC4*′ of the Product Rule.

Note that in terms of 'Cmprh″, notions analogous to those of being semantically null or universal may be defined, by using 'Cmprh″ in place of 'Cmprh' in the definitions of 'Null' and 'Univ'. And hence also a notion of semantical negation. These notions will be symbolized respectively by 'Null″, 'Univ″, and 'Neg″.

We can now prove the analogue *GRC5*′ of the Negative Rule as follows. From *TG9d*(III) we have that

(4) $\vdash$ SentFuncOne *c*,*d* . *a* = (*d abst c*) . *b* = (*d abst tilde c*) . SentFuncOne *e*,*d*′ :⊃: (SentFuncOne *c*,*d* . SentFuncOne *e*,*d*′ . Thm′ (*e hrsh c*) . SentFuncOne (*tilde c*),*d* . SentFuncOne *e*,*d*′ . Thm′ (*e hrsh tilde c*) :≡: Null′ (*d*′ *abst e*)),

and also, from *TG9e*(III), that

(5) $\vdash$ Hyps (of (4)) :⊃: (SentFuncOne *e*,*d*′ . SentFuncOne *c*,*d* . Thm′ (*c hrsh e*) . SentFuncOne *e*,*d*′ . SentFuncOne (*tilde c*), *d* . Thm′ (*tilde c hrsh e*) :≡: Univ′ (*d*′ *abst e*)).

Similarly

$\vdash$ Hyps (of *GRC5*′) :⊃: (*b* Cmprh′ *pee* . *a* Cmprh′ *pee* :≡: Null′ *pee*) . (*pee* Cmprh′ *b* . *pee* Cmprh′ *a* :≡: Univ′ *pee*).

Combining this with (4) and (5) we get that

$\vdash$ Hyps (of *GRC5*′) :⊃: *a* Neg′ *b*.

Here likewise additional clauses containing '*pee*' are needed. But these cannot hold on the hypotheses and hence may be added as disjuncts wherever needed.

In a somewhat similar way *GRC6*′ and *GRC7*′ may be proved.

GRC8′ is merely that

$\vdash$ (E*c*)(*a* Clsr′ *c* . LogAx *c*) . Vbl *b* :⊃: Univ′ (*b abst a*).

But this is immediate, using the syntactical law that

$\vdash$ (LogAx a .v. (Ed)(a Clsr d . LogAx d)) . Fmla c :$\supset$: Thm′ (c *hrsh* a).

By putting 'Cmprh'' for 'Cmprh' throughout the primitive expansion of *GRC10*, we gain *GRC10′*.

GRC10′. $\vdash$ SentFuncOne b,a . Vbl c . Univ′ (a *abst* b) :$\supset$: Univ′ (a *abst* c *qu* b).

This is provable as follows. First we note that

(6) $\vdash$ Hyps (of *GRC10′*) . SentFuncOne e,e' . d = (e' *abst* e) :$\supset$: Thm′ (e *hrsh* b).

Similarly

$\vdash$ Hyps (of (6)) :$\supset$: Thm′ (*tilde* e *hrsh* b).

Hence using *TG9e*(III) and *TG7a*(III),

$\vdash$ Hyps (of (6)) :$\supset$: Thm′ (c *qu* b),

and hence

(7) $\vdash$ Hyps (of (6)) :$\supset$: Thm′ (e *hrsh* c *qu* b),

and

(8) $\vdash$ Hyps (of *GRC10′*) :$\supset$: Thm′ (*pee*$\frown$*ex* *hrsh* c *qu* b).

Combining now (7) and (8) and using *TG8h*(III) we have that

$\vdash$ Hyps (of *GRC10′*) . (SentFuncOne e,e' . d = (e' *abst* e) :v: d = *pee*) :$\supset$: (SentFuncOne e,e' . d = (e' *abst* e) . SentFuncOne (c *qu* b),a . Thm′ (e *hrsh* c *qu* b)) .v. (d = *pee* . SentFuncOne (c *qu* b),a . Thm′ (*pee*$\frown$*ex* *hrsh* c *qu* b)).

From this we immediately get *GRC10′*.

We know from the material above in (III,H) and (V,F) that L^* is both consistent and complete, in the senses defined. (Cf. *TF13*(V).) Hence we can prove within M that

(9) $\vdash \sim$(Ea)(Fmla a . Thm′ a . Thm′ (*tilde* a))

and

(10) $\vdash$ (a)(Sent a :$\supset$: Thm′ a .v. Thm′ (*tilde* a)).

On the basis of (9) we can prove *GRC9′* and on the basis of (10) we can prove *GRC12′*.

GRC9′ is in effect that

$\vdash$ Univ′ e . Null′ a' . (SentFuncOne b,a . e = (a *abst* b) :v: e = *pee*) . (SentFuncOne c,d . a' = (d *abst* c) :v: a' = *pee*) :$\supset$: $\sim e$ Cmprh′ a'.

We have then four cases to prove. First consider the formula

(11) 'SentFuncOne b,a . SentFuncOne c,d . (e)(PredCon e .$\supset$. (a *abst* b)

Cmprh′ *e*) . (*e*)(PredCon *e* .⊃. *e* Cmprh′ (*d abst c*)) :⊃: ∼(SentFunc-One *b*,*a* . SentFuncOne *c*,*d* . Thm′ (*b′ hrsh c*))′.

We can show that this holds in *M* in view of (9). Note that

(12) ⊢ SentFuncOne *b*,*a* . SentFuncOne *c*,*d* . (*e*)(*e′*)(SentFuncOne *e*,*e′* :⊃: Thm′ (*e hrsh b*) . Thm′ (*c hrsh e*)) :⊃: Thm′ ((*b vee tilde b*) *hrsh b*).

Also

⊢ Hyps (of (11)) :⊃: Hyps (of (12)).

Hence

⊢ Hyps (of (11)) :⊃: Thm′ *b*,

and also

⊢ Hyps (of (11)). Thm′ (*b hrsh c*) :⊃: (*e*)(*e′*)(SentFuncOne *e*,*e′* .⊃. Thm′ *e*).

But from this we have that

⊢ Hyps (of (11)) . Thm′ (*b hrsh c*) :⊃: Fmla *b* . Thm′ *b* . Thm′ (*tilde b*).

Hence

⊢ Hyps (of (11)) . Thm′ (*b hrsh c*) :⊃: (E*a*)(Fmla *a* . Thm′ *a* . Thm′ (*tilde a*)).

Therefore in view of (9), (11) is a theorem. In a similar way we prove that

⊢ Univ′ *pee* . Null′ *a′* :⊃: ∼*pee* Cmprh′ *a′*,
⊢ Univ′ *e* . Null′ *pee* :⊃: ∼*e* Cmprh′ *pee*,

and

⊢ Univ′ *pee* . Null′ *pee* :⊃: ∼*pee* Cmprh′ *pee*.

Combining these appropriately, we gain *GRC9′* itself.

In a similar way we can show that *GRC12′* holds. *GRC12′* is that

⊢ Sent *a* . Vbl *c* :⊃: Univ′ (*c abst a*) .v. Null′ (*c abst a*).

But from (10) above we have that

⊢ Sent *a* . Vbl *c* :⊃: Thm′ *a* .v. Thm′ (*tilde a*).

And from syntax,

⊢ Thm′ *a* .⊃. (*b*)(PredCon *b* :⊃: (E*c*)(E*d*)(SentFuncOne *c*,*d* . *b* = (*d abst c*) . Thm′ (*c hrsh a*)) .v. (*b* = *pee* . Thm′ (*pee*∩*ex hrsh a*)),

and

⊢ Thm′ (*tilde a*) .⊃. (*b*)(PredCon *b* :⊃: (E*c*)(E*d*)(SentFuncOne *c*,*d* . *b* = (*d abst c*) . Thm′ (*a hrsh c*)) .v. (*b* = *pee* . Thm′ (*a hrsh pee*∩*ex*)) .

Hence we have that

(13) ⊢ Sent *a* . Vbl *c* :⊃: (*b*)(PredCon *b* .⊃. (*c abst a*) Cmprh′ *b*) .v. (*b*) (PredCon *b* .⊃. *b* Cmprh′ (*c abst a*)).

But (13) is *GRC12′* itself.

GRC11′ is merely that

$\vdash$ Sent a . Vbl b . Vbl c . $\sim b = c$:$\supset$: (b *abst* a) Cmprh′ (c *abst* a).

The proof of this is immediate.

GRC13′ states that

$\vdash$ SentFuncOne a,b . b FV a . d SF^{c}_{b} a . $\sim b = c$:$\supset$: (b *abst* a) Cmprh′ (c *abst* d).

This we can prove within M if we note that

(14) $\vdash$ Vbl b . Vbl c . $\sim b = c$:$\supset$: Thm′ ($c \frown id \frown b$).

But also

(15) $\vdash$ SentFuncOne a,b . b FV a . d SF^{c}_{b} a . $\sim b = c$:$\supset$: Thm′ ($c \frown id \frown b$ *hrsh* (d *hrsh* a)),

and hence, in view of (14),

$\vdash$ Hyps (of (15)) :$\supset$: SentFuncOne a,b . SentFuncOne d,c . Thm′ (d *hrsh* a).

But from this we easily get *GRC13′*.

The proof of *GRC14′* is immediate.

We gain then an interpretation of 'Cmprh' within M of such a kind that *GRC1–GRC14* are provable. Any inconsistency to be gained by the use of *GRC1–GRC14* in addition to the rules of logic and of M can be transformed into an inconsistency with M taken as syntax language for the appropriate kind of L. This establishes the relative consistency of *GRC1–GRC14*.

This "proof" of relative consistency, like that of (V,G), is couched informally in a natural meta-meta-language. But it could very likely be formalized without difficulty using only restricted logical and semantical concepts.

CHAPTER X

INDIVIDUAL AND FUNCTIONAL CONSTANTS

In the development of the various semantical meta-languages above, we have considered only object-languages which contain no primitive individual or functional constants. But we have seen that object-languages containing primitive individual or functional constants can be easily transformed into languages without such, and conversely, in view of Russell's theory of descriptions (II,F). The development of non-translational semantics in particular has been facilitated by excluding object-languages containing primitive individual or functional constants. Nonetheless, many important object-languages do in fact contain one or more such constants as primitive. Such languages were informally described in Chapter II, and have been shown especially important in inquiries in mathematics and the philosophy of science. Further, all the meta-languages formulated above contain primitive individual (structural-descriptive) constants and one primitive functional constant. Hence it will be useful to consider in this chapter the changes needed in order to render the various semantical theories above applicable to object-languages or to meta-languages (taken as object-languages) containing one or more primitive individual or functional constants.

First we must consider the *syntactical readjustments* required when the object-language is allowed to contain primitive *individual* constants. This will occupy § A, and in § B, the *reformulation of* SM_D^L will be outlined. In § C, some restricted denotation relations of *higher degree* are defined. In § D the *reformulation of non-translational semantics* based on comprehension is sketched, and some *theorems* in the non-translational semantics of individual constants are proved in § E. In § F the syntactical and semantical theory of *functional* constants is sketched. Finally, in § G we consider again the question of *self-reference*, in connection with non-translational semantics. And we shall see that a non-translational semantical meta-language can be formulated in such a way as to contain consistently its own (non-translational) truth-concept.

If L contains one or more individual constants as primitive, the semantical meta-language based on 'Den' for L will be called $SM_D^{L'}$, and the non-translational meta-language for L will be called $SM_C^{L'}$.

These two meta-languages differ only slightly from SM_D^L and SM_C^L respectively.

A. Syntactical Readjustments. When we allow the object-language L to contain one or more individual constants as primitives, certain slight changes must be made in the rigorous syntactical description of L. The syntactical changes to be sketched in this section will be common to both $SM_D^{L\prime}$ and $SM_C^{L\prime}$.

Let

'ā', 'ā'', 'ā''', . . . ,

be the finite or denumerably infinite list of primitive individual constants of L as in Chapter II. Each constant is regarded as naming a single individual, distinct constants naming distinct individuals if desired. The informal formulation of L is precisely as in Chapter II. In the formal syntax of L one must add a structural-descriptive constant, '*ay*', to name 'ā', and *ay* is to be distinct from *vee*, *tilde*, etc. This must be provided for by adding appropriate clauses to *SynR1* (III,B). Likewise to *SynR2*, a clause must be added to the effect that *ay* is not the concatenate of any expressions. The meaning of *SynR3* must be extended slightly, as well as that of *SynR4*. Let these rules now be called *SynR1'–SynR4'*. Similarly *R1'–R8'*, *MP'*, *Gen'*, and *Abst'* must be extended slightly.

An expression may be regarded as an individual constant of L if and only if it is either *ay* alone or consists of *ay* followed by one or more accents.

'InCon *a*' abbreviates '($a = ay$ .v. (E*b*)(E*c*)(AcString *b* . $a = (ay \frown b)$))'.

This definition is needed only where L contains a denumerable number of primitive individual constants. If L contains only a finite number, one would then define 'InCon *a*' similar to the way in which 'PrimPredCon *a*' was defined in (III,D), by enumeration. But howsoever we define the notion of being an individual constant, any expression *a* of L which is either a variable or an individual constant will be called a *term*.

'Trm *a*' abbreviates '(Vbl *a* .v. InCon *a*)'.

The definiens of the definition of 'AtFmla *a*' (III,D) need be changed only slightly, simply replacing the clause

'Vbl *b* . Vbl *c* . Vbl *d* . Vbl *e* . Vbl *a*''

by

'Trm *b* . Trm *c* . Trm *d* . Trm *e* . Trm *a*''.

Likewise, in the definition in (III,D) of '*a* Abst *b*', the phrase 'Vbl *d*'

must be replaced by 'Trm d'. But note that no further changes need be made in the definition of 'Fmla a', once these preliminary changes have been made.

In (III,E) we defined 'a $\mathrm{SF1}^b_c$ d' to express that a is like d except for containing a free occurrence of b in place of just one free occurrence of c in d. Because of the presence of individual constants in L now, we wish also to allow b in this definition to be an individual constant. Hence let 'a $\mathrm{S1}^b_c$ d' be defined here just as 'a $\mathrm{SF1}^b_c$ d' is defined in (III,E), but in place of 'b' FrOcc_a b' read '(InCon b . b' Occ_a b :v: b' FrOcc_a b)'. One then defines 'a $\mathrm{S?}^b_c$ d' similar to the way in which 'a $\mathrm{SF?}^b_c$ d' is defined in (III,E). 'a $\mathrm{S?}^b_c$ d' expresses that a differs from d only in containing an individual constant b or free occurrences of the variable b in 0 or more places where there are free occurrences of the variable c in d. Finally, 'a S^b_c d' is defined to express that a differs from d only in containing an individual constant b or free occurrences of a variable b *wherever* there are free occurrences of the variable c in d.

'a S^b_c d' abbreviates '(a $\mathrm{S?}^b_c$ d . (Vbl b . c FV a :⊃: $b = c$). (InCon b . ⊃ . ∼c FV a))'.

Also we define 'a $\mathrm{S1'}^b_c$ d' to express that a differs from d only in containing an individual constant b or free occurrences of the variable b in just one place where there is a free occurrence of the variable c in *d or an occurrence of the individual constant c* in d. And then 'a $\mathrm{S?'}^b_c$ d' and 'a $\mathrm{S'}^b_c$ d' may easily be defined to express the notions obviously intended by the notation.

In the definiens of the definition of 'QuantAx1 a' (in (III,E)) one now replaces 'b SF^c_d e' by 'b S^c_d e'. And in the definiens of the definition of 'IdAx1 a' one replaces 'Vbl b' by 'Trm b'. In that of 'IdAx2 a', 'Vbl b . Vbl c' is to be replaced by 'Trm b . Trm c'. And finally, in the definiens of the definition of 'AbstAx a', let 'd $\mathrm{SF?}^b_c$ e' be replaced by 'd $\mathrm{S?'}^b_c$ e . Trm c . Vbl b'.

Any further changes needed in the syntactical definitions or theorems may easily be supplied. We now turn to the changes required in the *semantical* part of $SM^{L'}_D$.

B. Individual Constants and Multiple Denotation. In $SM^{L'}_D$ multiple denotation might be reconstrued so as to allow individual constants to denote. But this would lead to complications which can be circumvented in another way. No change need be made in the statement of the Semantical Rules, *DenR1* and *DenR2* (IV,D), although their meaning is now slightly altered, because of the presence of individual constants as primitives in L.

We can say that an expression a is a *proper name of* an object x if and only if a is an individual constant and the abstract $(ex \frown invep \frown ex \frown id \frown a)$ denotes x.

'a PrNm x' abbreviates '(InCon a . $(ex \frown invep \frown ex \frown id \frown a)$ Den x)'.

To see that this definition of 'PrNm' provides an intuitively justifiable notion of being a proper name of, we note especially the following three theorems.

TB1. $\vdash a$ PrNm x .$\supset$. InCon a.

The proof is immediate.

TB2. $\vdash a$ PrNm y, if 'y' is taken as any individual constant in the translational part of $SM_D^{L'}$ and 'a' is taken as the structural description of that constant.

Using *DenR1* and *R5'*, we know that

$$\vdash (ex \frown invep \frown ex \frown id \frown a) \text{ Den } y \,.\equiv.\, y = y,$$

where (etc., as in *TB2*). But by *R7'*, noting that

$$\vdash \text{InCon } a$$

on the hypothesis, we get *TB2*.

Finally we have the Principle of Uniqueness, that an expression is a proper name of at most one entity.

TB3. $\vdash a$ PrNm x . a PrNm y :$\supset$: $x = y$.

The proof is as follows. From syntax we know that

(1) $\vdash \sim$InCon a . $a \neq ex$:$\supset$: $\sim$PredCon $(ex \frown invep \frown ex \frown id \frown a)$.

Hence, if we take 'a' as the structural description of an expression which is not *ex* or an individual constant, we know by (1) and *DenR2* that

$$\vdash \sim(ex \frown invep \frown ex \frown id \frown a) \text{ Den } y,$$

and hence,

(2) $\vdash$ InCon a . $(ex \frown invep \frown ex \frown id \frown a)$ Den y . $(ex \frown invep \frown ex \frown id \frown a)$ Den z :$\supset$: $y = z$.

Also if we take 'a' as the structural description of an individual constant 'y', using *DenR1* we have that

$$\vdash (ex \frown invep \frown ex \frown id \frown a) \text{ Den } x \,.\equiv.\, x = y.$$

Using *TE5*(II), an identity law, and *TE2*(IV), we have then that

(3) $\vdash$ InCon a . $(ex \frown invep \frown ex \frown id \frown a)$ Den x . $(ex \frown invep \frown ex \frown id \frown a)$ Den z :$\supset$: $x = z$,

taking 'a' as the structural description of an individual constant. Combining (2) and (3) and using *SynR4'*, we get the theorem.

These three laws should clearly hold on intuitive grounds of any notion of proper name in L that might be proposed. And all further properties of proper names in L that are desirable seem to be forthcoming from the definition given. The theory of the semantics of individual constants is thus merely the theory of denotation of expressions for certain unit virtual classes.

We shall also wish to say that a proper name is *subsumed* in an abstract. To allow this, we can define

$$\text{'}a \text{ Sub}^{\text{o}}\ b\text{' as '}(x)(a \text{ PrNm } x \mathrel{.\supset.} b \text{ Den } x)\text{'}.$$

(It might be thought that the definition of 'Sub' in (IV,C) could be used here. But if a is an individual constant, then according to that definition, a is subsumed in all b, because of *DenR2*. This rule states that

$$\vdash a \text{ Den } x \mathrel{.\supset.} \text{PredCon } a.$$

But

$$\vdash \text{InCon } a \mathrel{.\supset.} \sim\text{PredCon } a.$$

Therefore

$$\vdash \text{InCon } a \mathrel{.\supset.} \sim a \text{ Den } x,$$

and hence

$$\vdash \text{InCon } a \mathrel{.\supset.} (x)(a \text{ Den } x \mathrel{.\supset.} b \text{ Den } x).$$

Where a is an individual constant, we must therefore use the relation Sub^{o} in place of Sub.)

We now go on to the truth-concept. Under what circumstances is a sentence containing an individual constant to be regarded as *true*? If it is an atomic universal sentence or a non-atomic universal sentence, the definition can be given precisely as above (V,A) in (I) and (II).

But consider a sentence a of L which is *not* a universal sentence, e.g., 'Pā'. Under what circumstances can we say this is true? It is true if there is an expression b which denotes just those objects which have P, and an expression c which is a proper name of ā and c is subsumed in b in the sense just defined. Or, more generally, suppose a is a sentence but not a universal sentence. (L contains such a sentence because L contains primitively an individual constant.) We can then say that a is true provided a contains an occurrence of an individual constant b, there is a variable d as a free variable of some c where a differs from c only in containing b wherever there are free occurrences of d in c, and b bears Sub^{o} to the abstract $(d \frown invep \frown c)$.

'$\text{Tr}_3\ a$' abbreviates '$(\text{Sent } a \mathrel{.} \sim\text{UnivSent } a \mathrel{.} (\text{E}b)(\text{E}c)(\text{E}d)(\text{InCon } b \mathrel{.} d \text{ FV } c \mathrel{.} a\ \text{S}^b_d\ c \mathrel{.} b \text{ Sub}^{\text{o}}\ (d \frown invep \frown c)))$'.

The notion 'Tr_3' can perhaps best be made clear by an example.

Suppose a is a non-universal sentence of L '---ā---', containing the individual constant 'ā'. We then pick out some variable, say 'x', and form the sentential function '---x---' such that a differs from it only in containing 'ā' wherever there are free occurrences of 'x' in '---x---'. We then form the abstract '$x\ni$---x---' and require that 'ā' bear Sub^0 to '$x\ni$---x---'. A sentence such as 'Russell is mortal', e.g., is true in an appropriate formalism just where 'Russell' bears Sub^0 to '$x\ni x$ is mortal'.

The definition of '$\mathrm{Tr}_3\ a$' is to supplement definitions of '$\mathrm{Tr}_1\ a$' and '$\mathrm{Tr}_2\ a$', so that

'Tr a' abbreviates '($\mathrm{Tr}_1\ a$ .v. $\mathrm{Tr}_2\ a$ .v. $\mathrm{Tr}_3\ a$)'.

We gain then in $SM_D^{L\prime}$ a completely general predicate for the truth of the sentences of L.

The adequacy of 'Tr' may easily be proved by the Second Adequacy Rule of (V,C) and an appropriate Adequacy Rule for 'Tr_3'.

Several alternative definitions of truth can be given as in (V,B), including one in terms of vacuous abstracts. The definition in terms of vacuous abstracts has been used throughout in the development of truth theory and we shall continue to use it in the sequel. But the definition of 'Tr_3' given here is of interest in bringing out some of the semantical properties of individual constants. Moreover, the equivalence of the notion embodied in 'Tr_3' with that embodied in 'Tr' as defined in terms of vacuous abstracts and as applied exclusively to non-universal sentences containing individual constants, can easily be established using the Adequacy Rules.

The further development of $SM_D^{L\prime}$ closely parallels that of SM_D^L, and the proof of the consistency of $SM_D^{L\prime}$ relative to that of L plus the syntactical axioms is essentially that given in (V,G).

C. Relational Abstraction. An interesting feature of all the semantical theories developed above based on multiple denotation is that the only kinds of expressions of L that are needed as denoting expressions are *one-place* predicates. Multiple denotation is taken as a *two-place* semantical relation between one-place predicates of L and the objects of L. The relational symbols of L are then not regarded as denoting, so that there is no need to introduce relations of multiple denotation of higher degree. It might have seemed that such relations would be essential for handling the semantical properties of the relations of L. E.g., we might wish to be able to say that 'P'' denotes (in some sense) the objects x and y (taken in this order) provided P'xy. We should need then within the meta-language a *three-place* denotation relation. And to handle the triadic relations of

L, we should need within the meta-language a four-place denotation relation. And so on. The interesting fact is that no such denotation relations of higher degree have been required. We have been able to define in full generality the semantical truth-concept and various further semantical notions depending on it, using only the simple dyadic denotation relation. For these purposes at least it is clear that denotation relations of higher degree can be dispensed with altogether.

Nonetheless, denotation relations of higher degree are of some interest on their own account. And certain restricted relations of this kind are definable within $SM_D^{L\prime}$. The presence of individual constants in L and also in the meta-language is essential for this, and hence these relations cannot be introduced within SM_D^L, but only within $SM_D^{L\prime}$. First we must presuppose the method for handling relational abstraction within L, suggested at the end of (II,D). A restricted kind of three-place denotation relation is then definable as follows.

'a Den z,w' abbreviates 'Tr d', where 'a' is taken as the structural description of some two-place abstract '$xx'\varepsilon$---x---x'---' containing no free variables, 'z' and 'w' are taken as any individual constants of which 'b' and 'c' are respectively the structural descriptions, and 'd' is taken as the structural description of the sentence which results from '---x---x'---' by substituting the constants b and c respectively for all free occurrences of the variables 'x' and 'x''.

In a similar way one can define

'a Den x,y,z',

where (etc., about 'a', 'x', 'y', and 'z'). And so on. Note that we do not define

'a Den x,y' or 'a Den x,y,z'

for *variable* 'a', 'x', 'y', and 'z', but only where these are constants of appropriate kinds.

The utility of these definitions is perhaps somewhat limited. And in fact it is not clear that these more general kinds of denotation are needed in semantics. But if they should be needed, these definitions show that, within $SM_D^{L\prime}$, restricted forms of such denotation can be achieved.

D. Individual Constants and Comprehension. Let us now consider the changes required in the meta-language SM_C^L when L contains one or more primitive individual constant or constants. The syntactical changes required have already been outlined in § A

above, but *SynR4'* is dropped and *TG4* and *TG13* of Chapter III are taken in its stead, as in (VIII,D). 'Cmprh' is here the sole semantical primitive, so that $SM_C^{L'}$ is non-translational.

The truth-concept for atomic or non-atomic universal sentences (of L) containing no individual constants may be given as in (VIII,C). '$\mathrm{Tr}_1\, a$' and '$\mathrm{Tr}_2\, a$' may be presupposed as defined there. A relation akin to $\mathrm{Sub}^{\mathrm{o}}$ may also be introduced here so that '$\mathrm{Tr}_3\, a$' may be defined within $SM_C^{L'}$ as in § B above. 'Tr_3' applies only to sentences which are not universal and which therefore contain at least one individual constant. And then, as in § B,

'Tr a' can abbreviate '($\mathrm{Tr}_1\, a$ .v. $\mathrm{Tr}_2\, a$ .v. $\mathrm{Tr}_3\, a$)'.

Also various alternative definitions of 'Tr', including one in terms of vacuous abstracts, can be given here.

For the purpose of developing the semantical theory of individual constants in $SM_C^{L'}$, the following relation is more useful than $\mathrm{Sub}^{\mathrm{o}}$.

'a $\mathrm{Sub}^{\mathrm{oo}}$ b' abbreviates '(PredCon a . PredCon b . a Sub b :v: PredCon a . InCon b . a Sub ($ex \frown invep \frown ex \frown id \frown b$) :v: InCon a . PredCon b . ($ex \frown invep \frown ex \frown id \frown a$) Sub b :v: InCon a . InCon b . ($ex \frown invep \frown ex \frown id \frown a$) Sub ($ex \frown invep \frown ex \frown id \frown b$))'.

Also we let

'a $\mathrm{Cmprh}^{\mathrm{oo}}$ b' abbreviate 'b $\mathrm{Sub}^{\mathrm{oo}}$ a'.

In terms of '$\mathrm{Sub}^{\mathrm{oo}}$' or '$\mathrm{Cmprh}^{\mathrm{oo}}$', we can define various further notions akin to 'CoExt', 'DE', 'Unit', 'Com', 'Sum', etc., by putting '$\mathrm{Cmprh}^{\mathrm{oo}}$' for 'Cmprh' throughout the definientia. These notions may be symbolized respectively by '$\mathrm{CoExt}^{\mathrm{oo}}$', '$\mathrm{DE}^{\mathrm{oo}}$', '$\mathrm{Unit}^{\mathrm{oo}}$', etc. Some of these notions will be useful in stating the theorems of the next section.

The semantical rules of $SM_C^{L'}$ are those of SM_C^{L}, although their meaning is now slightly altered. Thus the relative consistency of these rules may be proved essentially as in (IX,G).

The theory of individual constants as formulated on the basis of any of the other non-translational primitives, such as 'Tr', 'Univ', 'Null', 'O' or '$\int$', is of course similar.

E. Some Semantical Theorems. We now turn to a few miscellaneous theorems in the non-translational semantics of individual constants.

First we list (without proof) a few syntactical theorems concerning individual constants.

TE1a. $\vdash$ InCon a . $c\ \mathrm{S}_d^a\ a'$. Sent a' :$\supset$: Sent c.

TE1b. $\vdash$ InCon a . $c\ \mathrm{S}'^a_b\ d$. Sent d . $\sim$UnivSent d :$\supset$: Sent c . $\sim$ UnivSent c.

TE1c. $\vdash c\ \mathrm{S}^a_b\ d\ .\supset.\ c\ \mathrm{S?}^a_b\ d.$

TE1d. $\vdash c\ \mathrm{S}'^a_b\ d\ .\supset.\ c\ \mathrm{S?}'^a_b\ d.$

TE2a. $\vdash c\ \mathrm{SF}^a_b\ d :\supset: c\ \mathrm{S}^a_b\ d\ .\ c\ \mathrm{S}'^a_b\ d.$

TE2b. $\vdash$ InCon a . $c\ \mathrm{S}^a_{d'}\ a'$. InCon b . $d\ \mathrm{S}^b_{d'}\ a'$. d' FV a' . SentFuncOne a',d' . $\sim$UnivSent $d :\supset:$ $\sim$UnivSent c.

As throughout, the truth-definition presupposed is that in terms of vacuous abstracts. The various theorems of Chapter IX, especially in § D, thus automatically become theorems of $SM^{L'}_C$.

Many of the semantical theorems to follow may easily be proved within a translational meta-language, using the appropriate Adequacy Rule. The proofs here, in a non-translational meta-language, are somewhat more difficult. Lacking any form of an Adequacy Rule, we must here seek altogether new proofs.

We first have a few theorems concerning CoExt^{oo}.

TE3a. $\vdash$ InCon a . InCon $b :\supset: a$ CoExt^{oo} b $.\equiv.$ Tr $(a\frown id\frown b)$.

By the definition of 'CoExt^{oo}',

$\vdash$ InCon a . InCon $b :\supset: a$ CoExt^{oo} b $.\equiv.$ $(ex\frown invep\frown ex\frown id\frown a)$ CoExt $(ex\frown invep\frown ex\frown id\frown b)$,

and hence by *TD22*(IX),

$\vdash$ InCon a . InCon $b :\supset: a$ CoExt^{oo} b $.\equiv.$ Tr $(ex\ qu\ (ex\frown id\frown a\ tripbar\ ex\frown id\frown b))$.

But by *TD10*(IX) and *TD15*(IX),

$\vdash$ InCon a . InCon $b :\supset:$ Tr $(ex\ qu\ (ex\frown id\frown a\ tripbar\ ex\frown id\frown b))$ $.\equiv.$ Tr $(a\frown id\frown b)$.

Hence *TE3a*.

TE3b. $\vdash$ InCon a . InCon b . a CoExt^{oo} b . $d\ \mathrm{S?}'^a_b\ c$. Sent $c :\supset:$ Tr $(c\ hrsh\ d)$.

Bearing in mind the syntactical changes of § A, we have, by *TD10* (IX), that

$\vdash$ InCon a . InCon b . $d\ \mathrm{S?}'^a_b\ c$. Sent $c :\supset:$ Tr $(a\frown id\frown b\ hrsh\ (c\ hrsh\ d))$,

and hence, by *TD7*(IX), that

$\vdash$ InCon a . InCon b . $d\ \mathrm{S?}'^a_b\ c$. Sent c . Tr $(a\frown id\frown b) :\supset:$ Tr $(c\ hrsh\ d)$.

We then get the theorem by *TE3a*.

We now note, without proof, a few lemmas concerning individual constants and unit predicate constants.

TE4a. $\vdash$ InCon a $.\supset.$ $\sim$Null $(ex\frown invep\frown ex\frown id\frown a)$.

TE4b. $\vdash$ Unit a . b Sub a . $\sim$Null $b :\supset:$ Unit b.

TE4c. $\vdash$ Unit a . c Neg $b :\supset: a$ Sub b $.\equiv.$ $\sim a$ Sub c.

TE4d. $\vdash$ InCon a .$\supset$. Unit ($ex \frown invep \frown ex \frown id \frown a$).

TE4e. $\vdash$ DE^{oo} a :$\equiv$: PredCon a .v. InCon a.

TE4f. $\vdash$ InCon a .$\supset$. a CoExt^{oo} ($ex \frown invep \frown ex \frown id \frown a$).

A fundamental theorem concerning individual constants is the following.

TE5. $\vdash$ InCon a .$\supset$. Unit^{oo} a.

The proof is not difficult, using some of the preceding lemmas.

Another fundamental theorem interrelates Sub^{oo} with truth. Suppose some individual constant a bears Sub^{oo} to a non-vacuous abstract '$x3$---x---'. Then the sentence we get by replacing the free occurrences of 'x' by a throughout '---x---' is true, and conversely.

TE6a. $\vdash$ InCon a . b FV d . c S^a_b d . SentFuncOne d,b :$\supset$: a Sub^{oo} ($b \frown invep \frown d$) .$\equiv$. Tr c.

The proof is by *TD21*(IX) and the principle that the theorems of L concerning identity, if sentences, are true. By *TD21*(IX),

$\vdash$ Hyps (of *TE6a*) :$\supset$: ($b \frown invep \frown b \frown id \frown a$) Sub ($b \frown invep \frown d$) .$\equiv$. Tr ($b$ *qu* ($b \frown id \frown a$ *hrsh* d)).

But

$$\vdash \text{Hyps} :\supset: \text{Tr}\ (b\ qu\ (b \frown id \frown a\ hrsh\ d)) .\equiv. \text{Tr}\ c,$$

by *TD10*(IX) and *TD15*(IX). Hence the theorem. A corollary of *TE6a* is that

TE6b. $\vdash$ InCon a :$\supset$: a Sub^{oo} *pee* .$\equiv$. Tr ($pee \frown a$).

We see this by recalling *TD23*(IX).

Several of the preceding theorems have given *necessary* conditions under which two individual constants can be said to be CoExt^{oo}. The following theorem gives a *sufficient* condition for such.

TE7. $\vdash$ InCon a . InCon b . $\sim a = b$. c FV a' . SentFuncOne a',e . c S^a_e a' . d S^b_e a' . a'' SF$^{e'}_e$ a' . $\sim e'$ FV a' . Tr c . Tr d . Tr (e *qu* e' *qu* ((a' *dot* a'') *hrsh* $e \frown id \frown e'$)) :$\supset$: a CoExt^{oo} b.

F. Functional Constants. We now sketch very summarily the adjustments required if the object-language L contains one or more primitive *functional* constants (functors).

Let

$$\text{`}\bar{\mathfrak{f}}\text{'}, \text{`}\bar{\mathfrak{f}}''\text{'}, \text{`}\bar{\mathfrak{f}}'''\text{'}, \ldots, \text{`}\bar{\mathfrak{f}}'''\ldots''$$

be the finite list of primitive functors of L, each of specified degree, as in Chapter II. We then add '*eff*' as a new structural-descriptive (individual) constant, in the syntax language M of L, to name '$\bar{\mathfrak{f}}$'. '$\bar{\mathfrak{f}}''$' then has '($eff \frown ac$)' as its structural description, and so on. The syntactical rules must be altered slightly so as to accommodate this new

constant. If L contains primitive individual constants also, it is *SynR1'–SynR4'* which must be altered; otherwise, *SynR1–SynR4* of Chapter III.

Obviously we may define

'FuncCon *a*'

by finite enumeration as needed. Suppose L contains just two primitive functional constants, '$\bar{f}$' of degree one, and '$\bar{f}$'' of degree two, and also some primitive individual constants. Then

'Trm *a*' may be defined as '(Vbl *a* .v. InCon *a* .v. (E*b*)(E*c*)((Vbl *b* .v. InCon *b*) . (Vbl *c* .v. InCon *c*) . (*a* = (*eff*⌒*b*) .v. *a* = (*eff*⌒*ac*⌒*b*⌒ *c*))))'.

The further changes required in the syntax of L may easily be supplied.

Functional constants do not denote in the sense of multiple denotation. But terms containing no free variables should be regarded as proper names of the corresponding objects. E.g., '$\bar{f}\bar{a}$' should be a proper name of the object $\bar{f}\bar{a}$. The notion of 'PrNm' of § B may therefore be extended as follows, in terms of 'Den'.

'*a* PrNm *x*' abbreviates '(Trm *a* . ~(E*b*)*b* FV *a* . (*ex*⌒*invep*⌒ *ex*⌒*id*⌒*a*) Den *x*)'.

The treatment of functional constants within non-translational semantics is similar, *mutatis mutandis*, to that of individual constants in § D above.

G. Self-Reference and Comprehension. We have noted in (IX,G), that the non-translational, semantical meta-language SM_C^L provides a consistent basis for the semantics of any first-order object-language L, on the assumption that the underlying syntax is consistent. And similarly for $SM_C^{L'}$, if L contains primitive individual constants. Because of its generality the method of constructing SM_C^L or $SM_C^{L'}$ can even provide *in a certain sense* a consistent semantics of itself. We have noted above, in (V,H), that the translational meta-languages based upon multiple denotation, or indeed upon the other primitives, cannot with consistency contain their own semantics. If such semantical meta-languages are consistent they cannot contain their own semantical truth-concepts. But a non-translational semantical meta-language is not only consistent relative to elementary syntax, as we have seen; it also can be formulated so as to contain its own semantical truth-concept.

To see this we construct a semantical formalism as follows. The syntax to be used is a variant of that of (V,H–I). The primitive

symbols, excluding for the moment the structural-descriptive constants, are '(', ')', '∼', 'v', '=', 'ɜ', '⌒', 'Cmprh', and variables '*a*', '*b*', etc. To simplify the syntax let us suppose all variables now written '*a*', '*a*′', '*a*′′′', etc. The primitive symbols, excluding the structural-descriptive constants, are then just 10 in number, namely,

(1) '(', ')', '∼', 'v', '=', 'ɜ', '⌒', 'Cmprh', '*a*', and ' '.

As primitive structural-descriptive constants of this formalism we take now names of each of these primitive symbols. We let

(2) '*lp*' name '(', '*rp*' name ')', '*tilde*' name '∼', '*vee*' name 'v', '*invep*' name 'ɜ', '*id*' name '=', '*cnc*' name '⌒', '*cmp*' name 'Cmprh', '*ay*' name '*a*', and '*ac*' name ' '.

(Note that '*ay*' is used here as the structural description of the *variable* '*a*', not of the constant 'ā' as in the preceding sections of this chapter.)

Each of these 10 structural-descriptive constants is a primitive individual constant, and each such constant will in turn require a structural description. Hence we might let

(3) '*LP*' name '*lp*', '*RP*' name '*rp*', '*TILDE*' name '*tilde*', etc.

But even with the addition of this list, the syntactical vocabulary is still not complete, because each of the structural-descriptive constants '*LP*', '*RP*', etc., must in turn have a structural-descriptive name. And these new structural-descriptive names must in turn have structural-descriptive names, and so on. The total vocabulary needed therefore comprises an infinity of structural-descriptive constants.

To provide a notation for this infinity of constants, we can use accents to form new ones just as we do to form new variables. We can think of '*lp*′′', e.g., as the name of '*lp*′', '*lp*′′′' as the name of '*lp*′′', and so on. And similarly for names of the other constants. In this way the constants of list (3) are provided for, so that '*LP*' may be defined as '*lp*′′', and so on. The total primitive vocabulary of the semantical meta-language consists then only of the symbols of (1) and of the structural-descriptive constants of list (2). The variables '*a*', '*a*′′', '*a*′′′', etc., range over the expressions of the very meta-language of which they are variables.[1]

This formalism we shall refer to as SSM_C, the (non-translational) semantical meta-language for itself based on comprehension.

The rules of SSM_C are essentially as in (VIII,D). The semantical

[1] In place of '=' and '⌒' here or in Chapter III we could if desired take a relation similar to Quine's M-relation as a primitive. See *Mathematical Logic*, pp. 287–90. But we should still have the problem of characterizing M axiomatically.

rules comprise adaptations of *GRC1–GRC13*. There is no need for an adaptation of *GRC14*, because SSM_C contains no primitive *one*-place predicate constants.

Various Boolean notions may be defined within SSM_C as in (VIII,B).

For each non-logical, i.e., syntactical or semantical rule of SSM_C, we can define within SSM_C a Specific Hypothesis of Comprehension. E.g., corresponding to *SynR1*, that

$$\vdash \sim lp = rp \,.\, \sim lp = tilde \,.\, \ldots \,.\, \sim ay = cmp,$$

we can define

'*SHC1*' as '$(a)(b)(\text{Vbl } b \,.\, a = (tilde\ LP \frown id \frown RP\ dot\ tilde\ LP \frown id \frown TILDE\ dot \ldots dot\ tilde\ AY \frown id \frown CMP) :\supset: \text{Univ } (b \frown invep \frown a))$'.

And corresponding to *GRC2*, e.g., or rather to a closure of the formula stipulated by *GRC2*,

$$\text{'}(a)(a')(a'')(a \text{ Cmprh } a' \,.\, a'' \text{ Cmprh } a :\supset: a'' \text{ Cmprh } a')\text{'},$$

we define

'*SHC7*' as '$(a)(b)(\text{Vbl } b \,.\, a = (ay\ qu\ ay \frown ac\ qu\ ay \frown ac \frown ac\ qu\ ((ay \frown cmp \frown ay \frown ac\ dot\ ay \frown ac \frown ac \frown cmp \frown ay)\ hrsh\ ay \frown ac \frown ac \frown cmp \frown ay \frown ac)) :\supset: \text{Univ } (b \frown invep \frown a))$'.

Within SSM_C various semantical notions are definable precisely as in (VIII,B–C), including a semantical truth-concept. The definitions of '*SHC1*' and '*SHC7*', just given by way of examples, presuppose that the semantical truth-concept is defined in terms of vacuous abstracts.

The proof of the relative consistency of SSM_C on the assumption that the rules of the underlying syntax are consistent, may be given as in (IX,G).

Throughout this section we have spoken of non-translational semantics only as formulated on the basis of a relation of comprehension. But obviously any alternative equivalent formulation could have been used equally well.

That the truth-concept for SSM_C is definable within itself might appear to contradict the result of Tarski mentioned above that, roughly speaking, the semantical truth-concept of a language is not definable within that language, if the language is consistent. But Tarski's result is concerned exclusively with systems of translational semantics for which the truth-concept must be adequate in essentially the sense of (V,C). That the truth-concept concerned be adequate is required in Tarski's argument, as we have seen in (V,H). Within non-translational semantics the condition of adequacy is abandoned in a certain sense, so that for systems of this kind an

essential step in Tarski's argument cannot be carried out. Hence there is no real conflict here with Tarski's result.

Because the semantical truth-concept of SSM_C is definable within SSM_C the consistency of SSM_C on suitable hypotheses can be proved within SSM_C along lines similar to those of (IX,F). For this proof use is made of the Specific Hypotheses of Comprehension for SSM_C, two of which were defined above. This in no way conflicts with the theorem of Gödel, mentioned in (V,I), to the effect roughly that if a given language-system L is consistent the statement that says that L is consistent is not provable within L. Such a statement is provable if at all only within a language containing additional assumptions. In the case of the proof of consistency of SSM_C one uses the Specific Hypotheses of Comprehension fundamentally. The deductive tools used in the proof of consistency are in effect considerably more powerful than just those of SSM_C without the Specific Hypotheses.

We see that SSM_C an in a certain sense safely refer to itself in the sense of containing its own syntax and semantics on suitable hypotheses.[1] Non-translational semantics is thus completely general in the sense that it can be formulated appropriately in a more or less uniform way to provide the syntax and semantics of any L, even of itself. Within non-translational semantics we achieve an extremely narrow, denumerable, first-order semantical meta-language for any first-order object-language of whatever complexity. We thus achieve in part the objective put forward in (I,H). A further simplification of non-translational semantics, in the direction of finitism, will be given in the next two chapters.

[1] For other kinds of systems which contain their own semantical truth-concepts, see F. B. Fitch, "A Basic Logic", *The Journal of Symbolic Logic*, 7 (1942), pp. 105–14; "Representations of Calculi", *ibid.*, 9 (1944), pp. 57–62; "A Minimum Calculus for Logic", *ibid.*, 9 (1944), pp. 89–94; "An Extension of Basic Logic", *ibid.*, 13 (1948), pp. 95–106; "The Heine-Borel Theorem in Extended Basic Logic", *ibid.*, 14 (1949), pp. 9–15; "A Further Consistent Extension of Basic Logic", *ibid.*, 14 (1950), pp. 209–17; "A Demonstrably Consistent Mathematics—Parts I and II", *ibid.*, 15 (1950), pp. 17–24, and 16 (1951), pp. 121–4. See also his *Symbolic Logic An Introduction*. Fitch's method imposes a restriction on the classical negation, whereas no such restrictions are needed here. See also J. R. Myhill. "A System Which Can Define Its Own Truth", *Fundamenta Mathematicae*, 37 (1950), pp. 190–2.

CHAPTER XI

INSCRIPTIONS AND MULTIPLE DENOTATION

All the semantical meta-languages considered in the foregoing chapters are classical in character, in the sense that the expressions of the object-language are regarded as sign-designs or shapes. For some years there has been interest in the possibility of an *inscriptional* syntax and semantics in which the expressions of the object-language are regarded exclusively as sign-events or inscriptions. This interest appears to have originated with the Polish logicians, but only very recently have any technical contributions been made in the construction of a systematic inscriptional theory.[1]

The present chapter and the next will be devoted to formulating two semantical meta-languages of an inscriptional kind. Inscriptions or sign-events will be taken as the only values for expressional variables. The first semantical meta-language will have as its sole semantical primitive a relation of multiple denotation. This meta-language will be akin to SM_D^L. The second inscriptional meta-language to be formulated will be akin rather to SM_C^L and here a relation of comprehension will figure as sole semantical primitive. The object-languages L throughout will be of the kind previously considered. But, to simplify, let us first suppose the L's to contain no primitive individual or functional constants. Then they may be added if desired and slight changes made in the syntactical and semantical material.

In SM_D^L and SM_C^L there is admitted a denumerable infinity of expressions as values for variables. Each primitive structural-descriptive term '*pee*', '*lp*', etc., is presumed to stand for a primitive symbol of the object-language L, and given any two structural-descriptive terms their concatenate is presumed to stand for an expression of L. This out-and-out admission of an infinity of expressions of L is in accord with the classical point of view. Because shapes, like numbers

[1] See N. Goodman and W. V. Quine, "Steps toward a Constructive Nominalism", *The Journal of Symbolic Logic*, 12 (1947), pp. 105–22; and R. M. Martin and J. H. Woodger, "Toward an Inscriptional Semantics", *ibid.*, 16 (1951), pp. 191–203. See also in this connection Tarski's *Der Wahrheitsbegriff*, pp. 290–1 and footnote 23.

or space-time points, are an abstract kind of thing never concretely met with in experience, the admission of an infinity of them seems legitimate.

When we consider sign-events or inscriptions we find a quite different situation. Here the values for expressional variables may be regarded as actual physical or phenomenal occurrences of some sort, as actual ink-marks or sounds discoverable within the spatio-temporal world. Here the explicit assumption of a denumerable infinity of inscriptions of L seems less natural. Possibly in fact the spatio-temporal order of the universe is finite in such a way that an at most finite number of inscriptions of L, however small, can exist. If this is the case, the assumption that there is an infinity of them is false. The actual situation is that we do not know whether there is a finite or an infinite number of actual inscriptions. Hence we shall wish to frame an inscriptional syntax and semantics for L in such a way as to assume neither one nor the other. The refusal to assume an infinity of expressions necessitates a rather considerable change in the syntactical and semantical material above.

By an inscription of L, let us understand (roughly) any typographical character or ink-mark of L taken by itself, or any sequence of the typographical characters of L arranged in a certain order. To make the order explicit we can think of the constituent characters of an inscription arranged consecutively as on a printed line. A line can be of any finite length, providing only that it is not too long to fit into the spatio-temporal world. And given any two lines the result of putting one followed by the other is also a line provided it is not too long. All the characters admitted belonging to a given object-language L are intended to be of the same typographical font, so that no drastic differences in size need be taken into account. (These requirements are not strictly essential, but they help the intuitive picture.)

The inscriptions of L are to be regarded as in principle discernible, as perceptible in some sense or other, although not necessarily by any one person. It is not required that they be explicitly perceived, however — some inscriptions might be too long. The inscriptions of L are regarded as a kind of physical or phenomenal entity, some of which are the ordinary ink-marks of experience, the others of which are like those ink-marks although possibly not explicitly exhibited within experience.

In § A the *syntactical preliminaries* for a systematic theory of inscriptions relativized to L are presented. In § B some *elementary theorems* are proved, in § C the *definition of 'Fmla'* is given, and *that of 'Thm'*, in § D. Some *further syntactical theorems* are proved in § E. In

§ F the foundations of the *inscriptional semantics* of *L* based on multiple denotation are given, including a truth-definition. § G is devoted to a proof of *adequacy* of this truth-definition. An *alternative definition* of truth in terms of vacuous abstracts is given in § H. Some of the material of this chapter and the next may appear to repeat material above. There are important differences of detail, however, and some ostensible repetition is therefore essential.

Reasons for philosophical interest in inscriptional syntax and semantics will be given in Chapter XIII. These are connected primarily with finitism and nominalism. But there are also other reasons for interest in inscriptions. To give an adequate semantical description of a language containing one or more of the so-called *egocentric particulars,* words such as 'I', 'here', 'this', and so on, reference to inscriptions seems essential.[1] Different inscriptions of these words have different semantical reference. Also inscriptions are intimately connected with phenomenalist theories of knowledge, such as that presented in Goodman's *The Structure of Appearance.* And finally, inscriptions are important for certain kinds of studies in empirical linguistics.[2]

Because so little work has been done upon inscriptional theory to date, the material of this and the next chapter is put forward somewhat tentatively. The *syntactical* groundwork has been laid by Goodman and Quine, and a first, rough attempt is made here to axiomatize what appears to be essentially their theory. Their work consisted merely of a sequence of definitions, the resulting system being only partially formalized (in the sense of Chapter I). But because axioms and definitions are so closely intertwined in any formalized language, it is important here to give a full specification of the axioms. We attempt also to build a semantics upon the basis of this syntax. These chapters thus contain a kind of tentative summary of the work which has been done on inscriptions to date. It is hoped they will be useful to others who may wish to improve upon the various theories, give more economical axioms, clarify their metaphysical or epistemological foundations, etc.

A. Syntactical Preliminaries. We first consider the syntactical readjustments which must be made in the material of Chapter III. These readjustments will be common to the two inscriptional semantical meta-languages to be constructed. The major one has already

[1] See, e.g., B. Russell, *An Inquiry into Meaning and Truth* (Allen and Unwin, London: 1940), Chapter VII.

[2] See, e.g., N. Chomsky, "Systems of Syntactic Analysis", *The Journal of Symbolic Logic,* 18 (1953), pp. 242–56.

been indicated, namely, that an infinite number of inscriptions need not be taken as values for variables. The inscriptional syntactical meta-language now to be formulated will be called *IM*.

IM is a very restricted first-order meta-language. Its logical primitives are as above (in III), i.e.,

'v', '∼', '(', ')', '⊃',

but with '=' omitted, this symbol being definable, as we shall see. As variables of the meta-language we use '*a*', '*b*', etc., with or without primes or numerical subscripts. The rules *R1–R6*, *MP*, *Gen*, and *Abst* of (II,A and D) are the logical rules of *IM*.

The object-languages *L* are to be taken as in Chapter III. We make, however, a slight change in the treatment of abstraction in *L*. In place of '⊃' we shall write ')', and therefore

'$x)A$'

in place of

'$x \supset A$'.

No confusion will result from using the isolated right parenthesis in this way. In *IM* we can use '⊃', but in *L* we use ')' instead. The reason for this slight change is in connection with the existence of inscriptions of *L*, as will be evident in a moment.

We now turn to the structural-descriptive symbols of *IM*. Here '*rp*', e.g., cannot be regarded as standing for ')'. The reason is that our informal use of quotation marks throughout has been the classical one, so that the sign ' ')' ' stands in effect for the class of all inscriptions shaped like a right parenthesis. Such classes are not values for variables within *IM*, as we have already noted. We might think then '*rp*' should stand for some specific right-parenthetical inscription picked out in advance as a standard. We could then say that an inscription *a* is a right parenthesis of *L* if *a* is sufficiently similar to *rp*. But for this we should need in *IM* the phrase 'sufficiently similar to'. Rather than to introduce this, we may regard '*rp*' and its like as *primitive one-place predicates* of *IM*. Because '*rp*' is to play the role of a primitive predicate, we shall capitalize the '*r*', in accord with the convention adopted in (III,D). 'Rp *a*' is to be significant here and is to be read 'the inscription *a* is a right parenthesis'. In a similar way, let

'Lp *a*' read '*a* is a left parenthesis',
'Vee *a*' read '*a* is a *vee*',
'Tilde *a*' read '*a* is a *tilde*',
'Id *a*' read '*a* is an identity-symbol',
'Ac *a*' read '*a* is an accent',

'Pee a' read 'a is a *pee*',

and

'Ex a' read 'a is an *ex*'.

'Lp', 'Rp', . . . , and 'Ex' are to be primitive, one-place, structural-descriptive predicate constants of *IM*.

In addition to these eight primitive predicate constants, we need concatenation as well as two other syntactical notions. We might, as in *M*, take '$\frown$' as a primitive functor. But '$(a \frown b)$' would then always be presumed to be a *term* of *IM*. Suppose a and b are each inscriptions of very great length. It might so happen that $(a \frown b)$ would be too long to fit into the spatio-temporal world! In this event, it would be the case that $\sim(\mathrm{E}c)(a \frown b) = c$, which would contradict the analogue of *TC1* of Chapter III that

$$\vdash (a)(b)(\mathrm{E}c)c = (a \frown b).$$

Of course we do not know yet whether this analogue is to hold in *IM*. But if we assume that every inscription is self-identical (in a sense to be defined) and allow '$(a \frown b)$' always to be a term, then the analogue of *TC1* follows. Hence it seems better to handle concatenation in a different way.

Let 'a C b,c' express that the inscription a is the concatenate of inscriptions b and c. 'C' will be taken as a primitive relational symbol of *IM* in place of the functor '$\frown$'. 'a C b,c' is to be understood in such a way that given b and c as inscriptions, a consists of b followed by c, taking b and c in their normal orientation to one another. 'a C b,c' does not hold, e.g., where a consists of b followed by the inscription c upside down. This notion of an orientation of b and c relative to each other is packed into the meaning of 'C'. Also a notion of order is involved. If a C b,c then $\sim a$ C c,b. But the order involved in concatenation can be given in a great variety of ways. It may be explicitly exhibited, as, e.g., where b and c are perceptible and spatio-temporally contiguous in some sense. Or, 'a C b,c' can hold where the order of b and c is merely described or describable. A concatenate is not given just when its constituent characters are. The order in which the characters are to be taken must also be stipulated or given in some way or another.

Also concatenation is to be understood in such a way that if a C b,c then b and c have no typographical character in common. Of course b and c may contain characters which are *similar* to each other, both Lp's, or Vee's, or whatever, but they cannot both contain one and the same character. This requirement seems essential if we think of inscriptions as arranged like printed lines. The same character could not appear, say, as both the 2nd and 32nd letter

on a printed line. It follows from this requirement that there is no concatenate of a character with itself.

If there is an infinity of typographical characters of L then, as suggested above, all concatenates whatsoever are assumed to exist. If there is only a finite number, say k_0, then all inscriptions whose length is less than or equal to k_0 are assumed to exist. In other words, the concatenates of the finite number of typographical characters may be taken in all possible ways. The number of inscriptions of length k_0 is $k_0!$ (i.e., $k_0 \times (k_0 - 1) \times (k_0 - 2) \times \ldots \times 2 \times 1$). The number of inscriptions of length $k_0 - 1$ is $k_0 \times (k_0 - 1) \ldots \times k_0 - (k_0 - 1) + 1$; of length $k_0 - 2$, $k_0 \times (k_0 - 1) \times \ldots \times k_0 - (k_0 - 2) + 1$; and so on. ($k_0!$ is the number of permutations of k_0 things taken k_0 at a time. $k_0 \times (k_0 - 1) \times \ldots \times k_0 - r + 1$ is the number of permutations of k_0 things taken r at a time ($1 \leq r \leq k_0$).) The number of all inscriptions of L is obviously the sum of these numbers. The assumption that all such concatenates exist seems reasonable when one reflects upon the vast enormity of inscriptions of L available in the whole of space-time, past, present, and future, here, there, and everywhere!

Note that, in talking about *IM*, we have continued to speak in the classical way, i.e., to regard the expressions of *IM* as shapes, not inscriptions. This mode of speaking, which has been used throughout, is the familiar one. Were we to regard the expressions of *IM* as inscriptions, we should not have the convenience, e.g., of using the classical quotation marks. But if desired, we could regard the expressions of *IM* as inscriptions, as the material of this chapter will show.

For the definitions of the basic syntactical notions relative to L, two further syntactical relations are required as primitives in *IM*. Let 'a Part b' express that the inscription *a is a part of* the inscription b in the sense that every character of a is a character of b with b's left to right order preserved. E.g., an inscription like:

$$()x\mathrm{vv}$$

is a part of the inscription:

(1) $$(x)\mathrm{v}x{\sim}\mathrm{vv}.$$

But no inscription like:

(2) $${\sim}x($$

is a part of the inscription (1), even though every character of (2) is like a character of (1).

Identity of inscriptions is definable in terms of 'Part'.

'$a = b$' abbreviates '(a Part b . b Part a)'.

Let 'a Lngr b' express that the inscription *a is longer than b* in

the sense of containing a greater number of characters. 'Lngr' is required as an additional syntactical primitive.

The logical axioms of *IM* have already been described. The *Rules of Syntax* of *IM* comprise adaptations of *SynR1–SynR3* of *M*, together with rules of existence, rules concerning Part and Lngr, and a kind of rule of induction.

The first rules are existence rules, stipulating in effect that *at least* such and such a number of Lp's, Rp's, etc., are available.

InSynR1a. $\vdash$ (Ea_1) Ea_2)...(Ea_n)(Lp a_1 . Lp a_2 . ... . Lp a_n . $a_1 \neq a_2$. $a_1 \neq a_3$ . ... . $a_1 \neq a_n$. $a_2 \neq a_3$ . ... . $a_{n-1} \neq a_n$), where $n = k_1$ and k_1 is the number of Lp's if there is only a finite number of them, or $n = 1, 2, \ldots$ if there is an infinite number.

Similar rules are assumed concerning Rp, Vee, Tilde, etc. These we may call *InSynR1b–InSynR1h*, respectively. k_2 is the number of Rp's, k_3 of Vee's, etc., if finite. We may regard these numbers including k_1 to be very large indeed although finite. k_0 above, the total number of characters of *L*, is then just the sum of $k_1, \ldots, k_8$. Of course we may never be able to give actual numerical values for k_1, etc. For present purposes all that is required is that they be sufficiently large.

We might wish also to add rules that there are *at most* k_1 Lp's, k_2 Rp's, and so on, if there is only a finite number of such respectively. But such rules are not needed here.

The next rule states that no Lp is a Tilde, no Vee a Tilde, etc.

InSynR2. $\vdash$ (a)(~(Rp a . Lp a) . ~(Rp a . Vee a) . ~(Rp a . Tilde a) . ... . ~(Ay a . Ex a)).

The notion of being a *typographical character of L* can be defined as follows.

'Char a' abbreviates '(Lp a .v. Rp a .v. Vee a .v. Tilde a .v. Id a .v. Ac a .v. Pee a .v. Ex a)'.

The next rule then stipulates that no character is a concatenate.

InSynR3. $\vdash$ Char a .$\supset$. ~(Eb)(Ec)a C b,c.

The next rule gives the circumstances under which two concatenates are identical. This law is analogous to *SynR3* of *M*.

InSynR4. $\vdash$ a' C a,b . d' C c,d :$\supset$: $a' = d'$.$\equiv$. ($a = c$. $b = d$:v: (Ee)(b C e,d . c C a,e :v: a C c,e . d C e,b)).

Also we have the following rules concerning the relations Lngr, Part, and C.

InSynR5. $\vdash$ a Part b .$\equiv$. (c)(c Part a .$\supset$. c Part b).

InSynR6. $\vdash$ a Lngr b . b Lngr c :$\supset$: a Lngr c.

InSynR7. $\vdash \sim a$ Lngr a.
InSynR8. $\vdash a$ Part $b :\supset: b$ Part a .v. b Lngr a.
InSynR9a. $\vdash a = b \,.\, a$ Lngr $c :\supset: b$ Lngr c.
InSynR9b. $\vdash a = b \,.\, c$ Lngr $a :\supset: c$ Lngr b.
InSynR9c. $\vdash a = b \,.\, a$ C $c,d :\supset: b$ C c,d.
InSynR9d. $\vdash a = b \,.\, c$ C $a,d :\supset: c$ C b,d.
InSynR9e. $\vdash a = b \,.\, c$ C $d,a :\supset: c$ C d,b.
InSynR10. $\vdash a$ C $b,c \,.\supset.\, \sim a$ C c,b.
InSynR11. $\vdash a$ C $b,c :\supset: b$ Part $a \,.\, c$ Part $a \,.\, \sim a$ Part $b \,.$ $\sim a$ Part c.
InSynR12. $\vdash a$ C $b,c :\supset: a$ Lngr $b \,.\, a$ Lngr c.
InSynR13. $\vdash a$ C $b,c \,.\, d$ Part $b :\supset: \sim d$ Part c.
InSynR14. $\vdash$ Char $a \,.\supset.\sim$(Eb)a Lngr b.
InSynR15. $\vdash$ Char $a \,.\, b$ Part $a :\supset: a$ Part b.[1]

All these rules seem natural enough and help to make explicit some of the informal comments above.

It is convenient to have available also relations of concatenation of higher degree, definable as follows.

'a C b,c,d' abbreviates '(Ee)(a C $b,e \,.\, e$ C c,d)',
'a C b,c,d,e' abbreviates '(Ea')(a C $b,a' \,.\, a'$ C c,d,e)',

and so on.

We may now make explicit the assumptions concerning the existence of concatenates. Let k_0 be the number of typographical characters of L, if finite, as above. The following rule, the *Rule of Existence*, stipulates that all possible concatenates of these k_0 characters exist.

InSynR16. $\vdash$ Char a_1 . Char a_2 . ... . Char a_n . $a_1 \neq a_2 \,.\, a_1 \neq a_3 \,.\, \ldots$ $.\, a_1 \neq a_n \,.\, a_2 \neq a_3 \,.\, \ldots \,.\, a_2 \neq a_n \,.\, \ldots \,.\, a_{n-1} \neq a_n :\supset:$ (Eb)b C $a_1,\ldots,a_n$. (Eb)b C $a_2,a_1,a_3,\ldots,a_n$. (Eb)b C $a_1,a_3,a_2,\ldots,a_n$) . ... . (Eb)b C $a_n,a_{n-1},\ldots,a_1$.

The conclusion of this rule is to be understood as containing, for a given n, just $n!$ distinct conjuncts,each stating that there is a concatenate of the n characters taken in one possible order. If k_0 is in fact the number of typographical characters of L, then n is to be successively 2, 3, . . ., k_0. But if there is an infinity of characters, n is to be successively 2, 3, . . ., so that in this case all possible concatenates of finite length are assumed to exist.

Before stating the next rule, a *Rule of Shapes*, it is convenient to introduce the following definition. Let 'S' with a numerical subscript stand for any one of the primitive structural-descriptive predicates, i.e., for any one of 'Lp', 'Rp', 'Vee', 'Tilde', 'Ac', 'Id', 'Pee', or 'Ex'.

[1] The form of this rule was suggested by Mr. Andrew Wessel.

One then defines

'$(\boldsymbol{S}_1\boldsymbol{S}_2 \ldots \boldsymbol{S}_n)\ a$' as an abbreviation for '$(\mathrm{E}b_1)(\mathrm{E}b_2)\ldots(\mathrm{E}b_n)(\boldsymbol{S}_1\, b_1 \,.\, \boldsymbol{S}_2\, b_2 \,.\, \ldots \,.\, \boldsymbol{S}_n\, b_n \,.\, a \subset b_1, b_2, \ldots, b_n)$'. ($n = 2, 3, \ldots, k_0$, or $n = 2, 3, \ldots$)

According to this definition, e.g.,

'(Ex Ac Ex Tilde Rp) a'

expresses that a consists of an Ex concatenated with an Ac, with another Ex, with a Tilde, and finally with an Rp. This definition provides a convenient way of specifying the shape of long concatenates. Hereafter we can let '$\boldsymbol{X}$', with or without primes or numerical subscripts, designate any of the primitive structural-descriptive predicates or any of the predicates introduced by this definition. Such predicates will be called *shape predicates,* being specificatory of shapes. Also we shall speak of the *length* of $\boldsymbol{X}$ in an obvious way. E.g.,

'(Ex Ac Ex Tilde Rp)'

has the length 5, and in general

'$(\boldsymbol{S}_1\boldsymbol{S}_2 \ldots \boldsymbol{S}_n)$'

has the length n. k_0 here is assumed, as above for *InSynR16,* to be the total number of typographical characters of L, if there is a finite number of them. In this case there will be little use for shape predicates of length greater than k_0. However, if there is an infinity of typographical characters of L, then n here is successively to be 2, 3, 4, etc., so that shape predicates of all finite lengths are defined.

The Rule of Shapes, which is a kind of induction rule analogous to *SynR4* of (III,B), may now read as follows.

InSynR17. If $\vdash (a)(\boldsymbol{X}\, a \,.\supset.\, \boldsymbol{F}\, a)$, for all shape predicates $\boldsymbol{X}$ of length $\leqq k_0$ (k_0 as above), or for all shape predicates $\boldsymbol{X}$ whatsoever, then $\vdash (a)\boldsymbol{F}a$.

This rule states that if we can prove that every inscription which has a given shape has a given property, and if we can prove this for all shape predicates (in the finite case) of length $\leqq k_0$, then we can prove that this property holds of all inscriptions. If in fact there is an infinity of inscriptions, then this rule is a rule of infinite induction. If there is not such an infinity, then this rule is a kind of rule of enumeration, enabling us to infer from a finite number of statements a general statement about all inscriptions whatsoever.

B. Some Elementary Theorems. We prove now a few elementary theorems of inscriptional syntax.

First, we have from *InSynR5* that

TB1a. $\vdash a$ Part a,

and

TB1b. $\vdash a \text{ Part } b \,.\, b \text{ Part } c :\supset: a \text{ Part } c.$

Using these we can slowly build up the basic theorems of identity.

TB2a. $\vdash a = b \,.\, a \text{ Part } c :\supset: b \text{ Part } c,$

and

TB2b. $\vdash a = b \,.\, c \text{ Part } a :\supset: c \text{ Part } b.$

Also

TB3a. $\vdash a = a,$
TB3b. $\vdash a = b \,.\supset.\, b = a,$

and

TB4. $\vdash a = b \,.\, b = c :\supset: a = c.$

Hence by *InSynR9a–InSynR9e*, using induction in the meta-language (of *IM*),

TB5. $\vdash a = b \,.\, \boldsymbol{F}\, a :\supset: \boldsymbol{F}\, b.$

Also by *InSynR4* we have immediately the associative law for concatenation.

TB6. $\vdash c \text{ C } a,e \,.\, b \text{ C } e,d \,.\, a' \text{ C } a,b \,.\, d' \text{ C } c,d :\supset: a' = d'.$

Using *InSynR11*,

TB7. $\vdash a \text{ C } b,c :\supset: \sim a = b \,.\, \sim a = c.$

Also, by *InSynR13*,

TB8. $\vdash a \text{ C } b,c \,.\supset.\, \sim(\text{E}d)(d \text{ Part } b \,.\, d \text{ Part } c).$

Hence also

TB9. $\vdash a \text{ C } b,c \,.\supset.\, \sim b = c.$

By *TB9* and *TB7*, we have that

TB10. $\vdash \sim a \text{ C } b,b \,.\, \sim a \text{ C } a,b \,.\, \sim a \text{ C } b,a.$

An immediate consequence of *InSynR4* is the uniqueness law for concatenation.

TB11. $\vdash a \text{ C } b,c \,.\, d \text{ C } b,c :\supset: a = d.$

The law of existence, that

$$(a)(b)(\text{E}c)c \text{ C } a,b,$$

is not assumed. Nor is its negative. It might so happen that there are enough inscriptions to justify this law, but this we do not assume.

Another form of *InSynR10* is that

TB12. $\vdash a \text{ C } b,c \,.\, d \text{ C } c,b :\supset: \sim a = d.$

In view of *InSynR15*, we have that

TB13a. $\vdash$ Lp a . b Part a :$\supset$: Lp b,
TB13b. $\vdash$ Rp a . b Part a :$\supset$: Rp b,

and so on.

C. Definition of 'Fmla'.[1] We now turn to some definitions of inscriptional syntax, similar to those of (III,D). These definitions are preparatory to that of the notion of *formula*.

'a Bgn b' and 'a Ends b' are definable as in (III,D) but with 'b C a,c' and 'b C c,a' replacing '$b = (a \frown c)$' and '$b = (c \frown a)$' respectively in the definientia. But 'a Bgn b' is to be read here 'the *inscription a* begins the *inscription b*'. And similarly for the other relations. Also 'a Seg b' is definable here as in (III,D). Note that 'a Seg b' means that a is a *continuous* part of b, whereas 'a Part b' holds even where a is a *scattered* part. If a Seg b then a Part b for all a and b, but not conversely.

The notion of being a *string of accents* is definable here essentially as in *M*.

'AcString a' abbreviates '(b)(Char b . b Part a :$\supset$: Ac b)'.

An inscription a is a *variable* of L if and only if it is an Ex or is an Ex followed by an AcString.

'Vbl a' abbreviates '(Ex a .v. (Eb)(Ec)(Ex b . AcString c . a C b,c))'.

The notion of being a primitive predicate constant of L may be defined essentially as in Chapter III.

'PrimPredCon a' abbreviates '(Pee a .v. (Eb)(Ec)(Ed)(Ee)(Ea')(Pee b . Ac c . Ac d . Ac e . Ac a' : a C b,c .v. a C b,c,d .v. a C b,c,d,e .v. a C b,c,d,e,a'))'.

This definition and the next presuppose, as in Chapter III, that L contains just five primitive predicate constants and that 'P' is of degree 1, 'P'' of degree 2, 'P''' of degree 3, 'P'''' of degree 4, and 'P''''' of degree 5.

The notion of being an *atomic formula* of L is also easily definable.

'AtFmla a' abbreviates '(Eb)(Ec)(Ed)(Ee)(Ea')(Eb')(Ec')(Ed')(Ee')(Ea'')(Eb'')(Vbl b . Vbl c . Vbl d . Vbl e . Vbl a' . Id b' . Pee c' . Ac d' . Ac e' . Ac a'' . Ac b'' : a C b,b',c .v. a C c',b .v. a C c',d',b,c .v. a C c',d',e',b,c,d .v. a C c',d',e',a'',b,c,d,e .v. a C $c',d',e',a'',b'',b,c,d,e,a'$)'.

The next step in the construction of inscriptional syntax is to

[1] The content of §§ C–D in all essential respects is due to Goodman and Quine (*op. cit.*).

define the notion of *formula*, as one would expect from the considerations of Chapter III. The idea naturally suggests itself of using the method of framed ingredients. But we are not allowing here expressions of all finite lengths. Recall that, according to the method of framed ingredients, an expression *a* is a formula if and only if there exists an expression *b* of which *a* is a framed ingredient and such that every framed ingredient of *b* is either an atomic formula, or is the negation or quantification or etc. of prior ingredients of *b*. Suppose *a* is a formula of very great length. Taking expressions as inscriptions, for sufficiently long *a* there might exist no inscription *b* of the kind demanded by this definition. For this reason, the method of framed ingredients fails when we take expressions as inscriptions. Hence, the notion of formula must be defined in another way. For this a few further preliminary definitions are needed.

Two inscriptions *a* and *b* are *equally long* if and only if neither is longer than the other.

'*a* EqLng *b*' abbreviates '(∼ *a* Lngr *b* . ∼ *b* Lngr *a*)'.

And two inscriptions *a* and *b* are *similar* or *like* each other if and only if they are equally long and such that for any two equally long inscriptions *c* and *d*, if *c* begins *a* and *d* begins *b*, then *c* and *d* both end in Lp's, Rp's, Vee's, Tilde's, or etc.

'*a* Like *b*' abbreviates '(*a* EqLng *b* . (*c*)(*d*)(*c* EqLng *d* . *c* Bgn *a* . *d* Bgn *b* :⊃: (E*e*)(E*e*′)(*e* Ends *c* . *e*′ Ends *d* . (Lp *e* . Lp *e*′ :v: Rp *e* . Rp *e*′ :v: Vee *e* . Vee *e*′ :v: Tilde *e* . Tilde *e*′ :v: Ac *e* . Ac *e*′ :v: Id *e* . Id *e*′ :v: Pee *e* . Pee *e*′ :v: Ex *e* . Ex *e*′))))'.

An inscription *a* is a *quantification* of *b* if and only if *a* consists of a universal quantifier prefixed to *b*.

'*a* Quant *b*' abbreviates '(E*c*)(E*d*)(E*e*)(Lp *c* . Vbl *d* . Rp *e* . *a* C *c*,*d*, *e*,*b*)'.

This notion is similar to that defined in (III,D), but 'Quant' is used here in place of 'Gen'. 'Gen' is reserved for another usage below.

An inscription *a* is a *negation* of *b* if and only if *a* consists of a Tilde prefixed to *b*.

'*a* N *b*' abbreviates '(E*c*)(Tilde *c* . *a* C *c*,*b*)'.

And *a* is a *disjunction* of *b* and *c* where *a* consists of an Lp concatenated with *a* concatenated with a Vee, then with *b*, and finally with an Rp.

'*a* D *b*,*c*' abbreviates '(E*d*)(E*e*)(E*a*′)(Lp *d* . Vee *e* . Rp *a*′ . *a* C *d*,*b*,*e*, *c*,*a*′)'.

For the definition of *formula* to be given we need a notion a little

more strict than D. We need to require not only that *a* D *b,c* but also that the Vee involved is the *main* connective of *a* and that the beginning and ending parentheses of *a* are *mates*. In other words, we must also require that *b* and *c* contain exactly as many left as right parentheses and that this is true of no inscription (other than *a*) which begins *a*.

First, then, we need the notion of *containing the same number of left as right parentheses*. Under what circumstances will an inscription *a* have this property? If *a* contains no parentheses at all, it will have it. And also it will have it if we pick out the (scattered) part of *a* containing only left parentheses, and the part containing only right parentheses, and require that these be equally long.

'EqPar *a*' abbreviates '(∼ (E*b*)(*b* Seg *a* : Lp *b* .v. Rp *b*) .v. (E*b*)(E*c*) (*b* EqLng *c* . (*d*)(Char *d* .⊃ . (Lp *d* . *d* Seg *a* :≡: *d* Seg *b*)) . (*d*)(Char *d* .⊃ . (Rp *d* . *d* Seg *a* :≡: *d* Seg *c*)))'.

Now we can say that an inscription *a* is a *disjunction* of *b* and *c* in the *strict* sense, provided *b* and *c* both have the property EqPar, there is no *d* and *e* such that EqPar *d* and *a* C *d,e* (i.e., no beginning segment of *a* other than *a* has the property EqPar), and *a* D *b,c*.

'*a* D_s *b,c*' abbreviates '(EqPar *b* . EqPar *c* . ∼(E*d*)(E*e*)(EqPar *d* . *a* C *d,e*) . *a* D *b,c*)'.

Also an inscription *a* is said to result from *b* by *abstraction* as follows.

'*a* Abst *b*' abbreviates '(E*c*)(E*d*)(E*e*)(Vbl *c* . Vbl *d* . Rp *e* . *a* C *c,e*, *b,d*)'.

Recall that we are writing abstracts in *L* using ')' in place of 'ɜ'. Hence *a* results from *b* by abstraction if *a* consists of a variable followed by a right parenthesis followed by *b* followed by a variable.

We can now define the notion of formula in two steps. First, a *quasi-formula a* is either an atomic formula, a disjunction in the strict sense of some *b* and *c*, or a negation of some *b*, or a quantification of some *b*, or results from some *b* by abstraction.

'QuasiFmla *a*' abbreviates '(AtFmla *a* .v. (E*b*)(E*c*)(*a* D_s *b,c*, .v. *a* N *b* .v. *a* Quant *b* .v. *a* Abst *b*))'.

And then a *formula a* is a quasi-formula where every disjunction (in the strict sense) contained in *a* has as its two components quasi-formulae, every negation contained in *a* is the negation of a quasi-formula, every quantification contained in *a* is the quantification of a quasi-formula, and for every *b* and *c*, if *b* is contained in *a* and results from *c* by abstraction, then *c* is a quasi-formula.

'Fmla a' abbreviates '(QuasiFmla a . $(b)(c)(d)(b\ \mathrm{D}_s\ c,d$. b Seg a :⊃: QuasiFmla c . QuasiFmla d) . (b) (c) $((b$ N c .v. b Quant c .v. b Abst c) . b Seg a :⊃: QuasiFmla c))'.

Not all quasi-formulae are formulae, because, e.g., the components of disjunctions and negations need not be formulae. But the formulae here comprise just the inscriptions intended. We can see this as follows. First, a formula a must be a quasi-formula. We require the disjuncts of every (strict) disjunction which is a segment of a also to be quasi-formulae. Starting with the longest disjunct which is a segment of a, its two disjuncts must be quasi-formulae. If either of the two disjuncts in turn is a disjunction, its two disjuncts must be quasi-formulae. And so on. And similarly for negations or quantifications or abstractions. In this way we can continue until we arrive at atomic formulae. And hence a formula in the sense defined is a formula in the intuitive sense of being constructible from atomic formulae by means of a finite number of steps of negation, disjunction, universal quantification, or abstraction.

Let us note now why, in the definitions of 'QuasiFmla' and 'Fmla', we need the notion D_s rather than D. Suppose $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{C}$ are in fact forms of atomic formulae. Then, e.g., we should wish an inscription of the form

(1) '$((\boldsymbol{A}$ v $\boldsymbol{B})$ v $\boldsymbol{C})$'

to be a formula. But note that an inscription of this form is in fact the disjunction in the loose sense D of an inscription of the form

'$(\boldsymbol{A}$'

with one of the form

'$\boldsymbol{B})$ v $\boldsymbol{C}$'.

But neither of these components in turn is a quasi-formula. Therefore, if we use 'D' in place of 'D_s' in the definitions of 'QuasiFmla' and 'Fmla', inscriptions of the form (1) would fail to be formulae. ('$\boldsymbol{A}$', etc., are used here in the sense of Chapter II.)

Another abbreviation that will be needed is as follows.

'$a\ \mathrm{C}_s\ b,c$' abbreviates '(Ed)(Ee)(Ea')(d N b . e N c . $a'\ \mathrm{D}_s\ d,e$. a N a')'.

Note that '$a\ \mathrm{C}_s\ b,c$' is related to conjunction in the way in which '$a\ \mathrm{D}_s\ b,c$' is related to disjunction.

D. Definition of 'Thm'. We now give a definition of the notion of being a theorem. Here, likewise, we cannot make use of the method of framed ingredients, for reasons similar to those which prohibited its use in the definition of 'Fmla'. If we should define 'Thm a' to express that there is an inscription b of which a or an inscription like

a is a framed ingredient, every framed ingredient of which is either an axiom or is obtainable from prior framed ingredients by means of one of the rules of inference, we should be requiring that given the appropriate *a* such a *b* always exist. The notion of being a theorem is, however, definable in another way as follows.

First, we need the notion of being an axiom. This is definable by an adaptation of the method of Chapter III. For this we introduce an abbreviation.

'*a* MC *b*,*c*' abbreviates '(E*d*)(*d* N *b* . *a* D_s *d*,*c*)'.

An inscription *a* is a *material conditional* of *b* and *c* if and only if there is an inscription *d* which is a negative of *b* and *a* is a strict disjunction of *d* and *c*. We can now define what it means to say that an inscription *a* is a truth-functional axiom of the first kind.

'TFAx1 *a*' abbreviates '(E*b*)(E*c*)(E*d*)(E*a*′)(Fmla *b* . *b* Like *c* . *c* Like *d* . *a*′ D_s *b*,*c* . *a* MC *a*′,*d*)'.

In a similar way, one defines 'TFAx2 *a*', 'TFAx3 *a*', and 'TFAx4 *a*'.

Before defining the notion of being an axiom of quantification, we need to define the notions of *being a free variable of* and of *substitution*.

A variable *a* is a *free variable* of an inscription *b* if and only if *a* is a segment of *b* not followed by any additional accents in *b*, and is such that *a* does not occur in (is not a segment of) any segment of *b* which consists of a formula to which is prefixed a quantifier consisting of an Lp, a variable like *a*, and an Rp.

'*a* FV *b*' abbreviates '(Vbl *a* . *a* Seg *b* . (*c*)(*d*)(Ac *d* . *c* C *a*,*d* :⊃: ∼*c* Seg *b*) . (*a*′)(*b*′)(*c*′)(*d*′)(*e*′)(Lp *a*′ . *a* Like *b*′ . Rp *c*′ . Fmla *d*′ . *e*′ C *a*′,*b*′,*c*′,*d*′ . *e*′ Seg *b* :⊃: ∼*a* Seg *e*′))'.

The notion of substitution was defined in Chapter III using the method of framed ingredients. This method is not available here for reasons already indicated. We wish to define here what it means to say that a formula *a* differs from a formula *d* only in containing free variables *like b* wherever *d* contains free variables *like c*. Clearly *a* and *d* will be interrelated in this way if we require that some inscription which results from *d* by deletion of *all* free variables like *c* is like some inscription which results from *a* by deleting *some* free variables like *b*. Hence,

'*a* SF^b_c *d*' abbreviates '(Fmla *a* . Fmla *d* . Vbl *b* . Vbl *c* . (E*a*′)(E*b*′)(*a*′ Like *b*′ . (*c*′)(Char *c*′ :⊃: ((*d*′)(*d*′ Like *c* . *d*′ FV *d* :⊃: ∼*c*′ Seg *d*′) . *c*′ Seg *d* :≡ :*c*′ Seg *b*′)) . (*c*′)(Char *c*′ . (*d*′)(*d*′ Like *b* . *d*′ FV *a* :⊃: ∼*c*′ Seg *d*′) :⊃: (*c*′ Seg *a* .≡. *c*′ Seg *a*′)))))'.

That an inscription is a QuantAx1 is now definable.

'QuantAx1 a' abbreviates '(Eb)(Ec)(Ed)(Ee)(Ea')(Eb')(Ec')(Ec'') (Lp a' . Rp b' . d C a',c,b',b . c' Like c . e $SF^{c''}_{c'}$ b . a MC d,e)'.

In a similar way one can define 'QuantAx2 a', 'IdAx1 a', and 'IdAx2 a'. The definitions are similar to those of Chapter III. A further slight change must be made in the definition of being an axiom of abstraction, 'AbstAx a', because of the use in L of ')' in place of '$\supset$'. Then the notion of being a *logical axiom* of L may be defined essentially as in (III,E).

'LogAx a' abbreviates '(TFAx1 a .v. ... .v. AbstAx a)'.

And in a similar way one can define the notion of being a descriptive or *non-logical axiom of L*. Then let 'Ax a' symbolize that a is either a logical or a non-logical axiom of L.

An inscription a is an *immediate consequence* of inscriptions b and c if it is obtainable from b and c by means of Modus Ponens or from b by means of the Rule of Generalization. Hence,

'a IC b,c' abbreviates '(a MP b,c .v. a Gen b)',

where

'a MP b,c' abbreviates '(Ed)(Ee)(Ea')(a like d . c Like e . b Like a' : a' MC e,d .v. e MC a',d)',

and

'a Gen b' abbreviates '(Ec)(c Quant b . a Like c)'.

These notions are similar to the corresponding notions of Chapter III, but with the additional clauses concerning likeness.

A *proof* is, in general, it will be recalled, a sequence of formulae of such a kind that each formula in it is either an axiom or can be obtained from preceding formulae by means of the rules of inference. Each such formula can be called a *step*. A proof will usually be an inscription consisting of several steps. Let us say, more precisely, that a is a *step* of b, for any inscriptions a and b, provided that a is a formula and is a part of b, and provided that a is not part of another formula which in turn is a part of b.

'a Step b' abbreviates '((c)(Fmla c . a Part c . c Part b :$\supset$: $a = c$) . Fmla a . a Part b)'.

To say that a is not part of another formula which is a part of b is to say, obviously, that every formula c of which a is a part and which is a part of b is identical with a. Given the notion of being a step, we can go on to the notion of being a proof.

Consider any inscription b which contains as parts some steps of an inscription a, none of which are axioms. If this holds, then let some step of a which is a part of b be an immediate consequence of steps of

a which are not parts of b. Given a, let this hold for all b whatsoever. In this case we can say that the inscription a is a *proof*.

'Proof a' abbreviates '$(b)((\mathrm{E}c)(c \text{ Step } a \,.\, c \text{ Part } b) \,.\, (c)(\mathrm{Ax}\ c \,.\, c \text{ Step } a :\supset: \sim c \text{ Part } b) :\supset: (\mathrm{E}c)(\mathrm{E}d)(\mathrm{E}e)(c \text{ Step } a \,.\, c \text{ Part } b \,.\, d \text{ Step } a \,.\, \sim d \text{ Part } b \,.\, e \text{ Step } a \,.\, \sim e \text{ Part } b \,.\, c \text{ IC } d,e))$'.

We must now note that a proof a in this sense is in fact a proof in the sense of Chapter II, according to which a proof is a sequence of formulae $d_1, \ldots, d_n$ each of which is either an axiom or can be obtained from preceding steps by means of a rule of inference. To see this we first show that if a is a proof in the sense of this definition, then an order can be specified among the steps of a such that each step is an axiom or an immediate consequence of preceding steps. Given a, we first pick out those steps of a which are axioms. Suppose these are $d_1, \ldots, d_k$. Consider now a part b of a which consists of all those steps of a except $d_1, \ldots, d_k$. If a is a proof this part b will contain a step which is an immediate consequence of steps of a outside b. We can call this step d_{k+1}. Then we consider another part c of a consisting of all those steps of a other than $d_1, \ldots, d_{k+1}$. This part likewise will contain a step which is an immediate consequence of steps of a outside b. Call this step d_{k+2}. And so on, until all the steps of a are exhausted. In this way we achieve the sequence of formulae $d_1, \ldots, d_k, \ldots, d_n$, which is a proof in the intuitive sense.

Now, conversely, suppose we are given a sequence $d_1, \ldots, d_n$ of formulae constituting a proof in the intuitive sense, where each of $d_1, \ldots, d_n$ is a step of a. We can then show that a is a proof in the sense of the definition as follows. Consider any inscription b containing some steps of a but none which are axioms. From among these steps of a which are parts of b, pick out the one which is earliest in the assumed order. Suppose this is d_i, d_i must, assuming $d_1, \ldots, d_n$ to constitute a proof in the intuitive sense, be either an axiom or an immediate consequence of preceding steps. But it is not an axiom, being contained in b. Hence d_i is an immediate consequence of steps of a not in b. Therefore b contains at least one step of a which is an immediate consequence of steps of a outside b. We see then that a must be a proof in the sense of the definition, b being any inscription of L containing some steps of a but none which are axioms.

In terms of this notion of proof, we can define the notion of 'Thm' as follows.

'Thm a' abbreviates '$(\mathrm{E}b)(a \text{ Step } b \,.\, \text{Proof } b)$'.

So that an inscription a is a theorem if and only if it is a step of some proof.

In a similar way, we can define the notion of 'LogThm a', by altering the definition of 'Proof b' so as to include only logical axioms.

Also we may presuppose as defined any further syntactical notions that might be needed. In particular, 'Sent a', 'SentFuncOne a,b', 'PredCon a', 'AtUnivSent a', 'UnivSent a', etc., are to be defined now essentially as in (III,F), but with such slight changes as are needed. E.g., a is a sentential function of the one variable b now if and only if a is a formula, b a variable, if b is a free variable of a then every free variable of a is like b, and if b is not a free variable of a then a contains no free variable at all. Also we let

'QuantString a' here abbreviate '((Eb)(Ec)(Lp b . Rp c . (Ed)a C b,d,c . (a')(b')(c')(d')(Lp b' . Rp d' . a' C b',c',d' . a' Seg a :⊃: Vbl c' .v. (Ee)(Ee')(Ee'')(Rp e' . Lp e'' . e C e',e'' . e Seg c'))) . (b)(c)(d)(Lp b . Rp c . d C b,c :⊃: ∼d Seg a))'.

'a Clsr b' and 'a Clsr' b' may then be defined essentially as in (III,F).

E. Some Further Theorems. Before going on to inscriptional semantics, we pause to note without proof a few further syntactical theorems.

We note first that inscriptional analogues of the theorems of (III,C), giving elementary properties of the classical concatenation operation, are now provable in inscriptional syntax. A few of these analogues have already been proved in § B. In the present section, a few further such theorems will be listed, as well as some theorems which have no direct analogues in classical theory.

First we have an induction rule analogous to *TC3a*(III).

TE1. If ⊢ (a)(Char a .⊃. ***Fa***) . (a)(b)(c)(***Fb*** . ***Fc*** . a C b,c :⊃: ***Fa***), then ⊢ (a)***Fa***.

('***F***' is used here in the sense of (II,D), as throughout.) The proof is by *InSynR17*.

Next we have a few theorems concerning the notions defined in §§ C–D.

TE2a. ⊢ a Seg b .⊃. a Part b.

TE2b. ⊢ ∼(a)(b)(a Part b .⊃. a Seg b).

TE2c. ⊢ a Seg a.

TE2d. ⊢ a Seg b . b Seg c :⊃: a Seg c.

TE3a. ⊢ EqPar a . EqPar b . (c D a,b .v. c N a .v. c Quant a) :⊃: EqPar c.

TE3b. ⊢ EqPar b . a Abst b :⊃: ∼EqPar a.

TE4a. ⊢ Fmla a . Fmla b . c D a,b . d D_s a,b :⊃: c Like d.

TE4b. ⊢ Ax a .⊃. Fmla a.

TE4c. ⊢ Fmla a . Fmla b . c IC a,b :⊃: Fmla c.

We have also the following theorems concerning likeness.

TE5a. $\vdash a$ Like a.

TE5b. $\vdash a$ Like b . b Like c :⊃: a Like c.

TE5c. $\vdash a$ Like b .⊃. b Like a.

TE5d. $\vdash a$ Like b . c C a,e . d C b,e :⊃: c Like d.

TE5e. $\vdash a$ Like b . c C e,a . d C e,b :⊃: c Like d.

TE5f. $\vdash a'$ C a,b . d' C c,d . a Like c . b Like d :⊃: a' Like d'.

TE5g. $\vdash a'$ C a,b . d' C c,d . (Ee)(Ee')(Ea'')(Ed'')(e Like e' . a Like a'' . d Like d'' . b C e,d'' . c C a'',e') :⊃: a' Like d'.

TE5h. $\vdash a'$ C a,b . d' C c,d :⊃: a' Like d' .≡. ((a Like c . b Like d) .v. (Ee)(Ee')(Ea'')(Ed'')(e Like e' . a Like a'' . d Like d'' . b C e,d'' . c C a'',e') .v. (Ee)(Ee')(Eb'')(Ec'')(e Like e' . b Like b'' . c Like c'' . a C c'',e . d C e',b'')).

TE5i. $\vdash a$ Like b . Fmla a :⊃: Fmla b.

TE5j. $\vdash a$ Like b . Thm a :⊃: Thm b.

A very important principle is the following, whose role in inscriptional syntax is somewhat similar to that of the Rule of Framed Ingredients (*TG4* of III) within classical syntax. This principle, called the *Rule of Proof*, states that if (i) a is a step of some proof b, (ii) every axiom of L has such and such a property, and (iii) this property is closed under the relation of immediate consequence, then a also has the property.

TE6. $\vdash a$ Step b . Proof b . (c)(Ax c .⊃. $\boldsymbol{F}c$) . (c)(d)(e)($\boldsymbol{F}c$. $\boldsymbol{F}d$. e IC c,d :⊃: $\boldsymbol{F}e$) :⊃: $\boldsymbol{F}a$.

The proof of *TE6* is primarily by *InSynR17*.

Also we have the following theorems which may be useful below.

TE7a. $\vdash$ Thm a .⊃. Fmla a.

TE7b. $\vdash$ Thm a . Thm b . c IC a,b :⊃: Thm c.

TE8a. $\vdash$ AtUnivSent a .⊃. (Eb)(Ec)(Lp b . a C b,c).

TE8b. $\vdash$ Sent a :⊃: AtUnivSent a .v. (Eb)(Sent b . a N b) .v. (Eb)(Ec)(Sent b . Sent c . a D_s b,c).

TE8c. $\vdash$ Sent a . a D_s b,c :⊃: Sent b . Sent c.

F. Multiple Denotation and Truth.[1] We now turn to the inscriptional *semantical* meta-languages presupposing *IM* as their syntactical part. The first semantical meta-language for L will be called ISM^L_D. A relation of multiple denotation will be taken as sole semantical primitive. This relation is essentially the same as that

[1] The content of §§ F–G essentially follows Martin and Woodger, *op. cit.* But no assumption of an infinity of inscriptions is made here. This constitutes an important improvement over the Martin-Woodger treatment.

discussed in Chapter IV, but its domain here consists of inscriptions or sign-events, not sign-designs. '*a* Den *x*' now reads 'the inscription *a* denotes the object *x*'. And here, as in Chapter IV, the forms '*x* Den *x*'', '*x* Den *a*', and '*a* Den *b*' are regarded as meaningless.

ISM^L_D, as is evident from the preceding paragraph, is a translational meta-language. Hence it is to contain translations of all the primitive symbols of *L*. Here, as in Chapter IV, for convenience we use the very symbols of *L* as their own translations. ISM^L_D contains two styles of variables, like SM^L_D, the one taking the objects of *L* as values, the other, the expressions of *L* regarded as inscriptions. As in SM^L_D, the two kinds of objects here may, if desired, be regarded as constituting two mutually exclusive domains.

The following can be defined in ISM^L_D essentially as in (IV,C): '*a* Sub *b*', '*a* Cmprh *b*', '*a* CoExt *b*', 'Com *a*', 'Unit *a*', 'Null *a*', 'Univ *a*', '*a* Ext ***F***', '*a* Sum *b*,*c*', '*a* Prod *b*,*c*', and '*a* Neg *b*'.

We now turn to a truth-definition. We first define truth as applied to atomic universal sentences essentially as in (V,A). I.e., we let

(I) 'Tr_1 *a*' abbreviate '(AtUnivSent *a* . (E*b*)(E*c*)(Lp *c* . *a* C *c*,*b* . (*x*) *b* Den *x*)'.

Note that this definition will not conflict with the restriction concerning the length of admissible inscriptions. Because, given an inscription *a*, the required *b* is always forthcoming as a part of *a*, i.e., simply as the result of deleting the left-most left parenthesis of *a*. To have *b* always forthcoming in this way is the reason for using isolated right parentheses in place of inverted epsilons in writing abstracts in *L*. But if we require *b* to differ from *a* by deleting the left-most left parenthesis of *a* and then replace the left-most right parenthesis of *a* by some new kind of inscription of the form '∋', we would not be sure that the resulting *b* exists. *a* might be so long that all inscriptions of the form '∋' might already be contained in *a*! But as we have seen there is no need to use '∋' here within *L*. The definition (I) requires the *b* concerned to be universal, and of course holds only for atomic universal sentences.

If we try to supplement this definition by a definition similar to that of 'Tr_2 *a*' in (V,A), using conjunctive normal forms, we run into difficulty. Because given *a* of sufficient length, the conjunctive normal form might have to be too long to exist! Hence this form of definition must be rejected.

Next we might consider using prenex normal forms, as in the definition (III) of (V,B). But here too there might be difficulties concerning the existence of a prenex normal form for sufficiently long *a*. E.g., suppose *a* is of the form

(1) $$'(x')(\sim(x)\boldsymbol{A} \vee \boldsymbol{B})',$$

where $\boldsymbol{A}$ and $\boldsymbol{B}$ are in primitive notation and contain no quantifiers and 'x' does not occur freely in $\boldsymbol{B}$. And suppose a to be so long as to contain all the Tildes, e.g., in the whole of space-time. Written in primitive notation, one prenex normal form of a would be of the form

(2) $$'(x')\sim(x)\sim(\sim\boldsymbol{A} \vee \boldsymbol{B})'.$$

To write this we must have two more Tildes than are contained in a. Hence, even where a might exist, an inscription of the form (2) might not. And similarly for any other prenex normal form of a. (Again, '$\boldsymbol{A}$' and '$\boldsymbol{B}$' are used here essentially in the sense of Chapter II.)

Of course we could use the artificial device of requiring that all sentences of L be atomic universal sentences. In this event, (I) would provide a completely general definition of truth for L. But to require that all sentences of L be atomic universal sentences is somewhat *ad hoc*, and we should prefer not to do so if it can be avoided.

Let us therefore turn to another kind of truth-definition in which *shape properties* are regarded as the true things and not inscriptions. The disadvantage of this procedure is that we shall have to use a somewhat more complicated notation. But the advantage is that we shall be able to gain the effect of a truth-definition applicable to all actual sentences of L whatsoever regardless of length.

Let '$\boldsymbol{A}$', with or without subscripts or primes, designate hereafter any formula of ISM_D^L. We shall speak of a shape predicate $\boldsymbol{X}$ as the *structural description* of $\boldsymbol{A}$, where $\boldsymbol{A}$ is a formula from the translational part of the meta-language, under appropriate circumstances. E.g.,

$$'(\text{Lp Ex Rp Ex Id Ex})'$$

is the structural description of '$(x)x = x$'. Also we shall say that a shape predicate $\boldsymbol{X}$ is an *atomic universal sentence* contained in $\boldsymbol{X}'$, if $\boldsymbol{X}$ is the structural description of an $\boldsymbol{A}$ which is an atomic universal sentence contained in $\boldsymbol{A}'$, and $\boldsymbol{X}'$ is the structural description of $\boldsymbol{A}'$.

Bearing this syntactical notation in mind, we can proceed directly to the definition of the truth-concept. The definition is in two parts, analogous roughly to the two parts of the definition in (V,A). The first definition is concerned with atomic universal sentences, the second with universal sentences which are non-atomic. The definition of '$\text{Tr}_1\, a$' ((I) above) is presupposed.

(IIa) '$\text{TS}_1\, \boldsymbol{X}$' abbreviates '$(\text{E}a)(\boldsymbol{X}\, a.\ \text{Tr}_1\, a)$'.

(IIb) '$\text{TS}_2\, \boldsymbol{X}$' abbreviates '$(\text{E}a)(\boldsymbol{X}\, a\ .\ \text{UnivSent}\ a\ .\ \sim\text{AtUnivSent}\ a\ .\ \boldsymbol{A})$', where $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ are just the constituent atomic universal

sentences of X (taken in left to right order of occurrence in X), X is the structural description of a formula A' of the translational part of ISM_D^L, X_k is the structural description of A_k ($1 \leq k \leq n$), and A differs from A' only in containing '$\mathrm{TS}_1\ X_k$' wherever A' contains A_k. The n here may be chosen as any integer ≤ 2 but no greater than some integer m_0 where m_0 is the number of atomic universal sentences (counting repetitions) occurring in the longest inscription of L which is a sentence. Of course it is very unlikely that we could ever give a specific numerical value for m_0. But if the universe is finite in the appropriate sense, then there is such a number. If the universe is not finite, then let $n = 2, 3, \ldots$.

(IIa) and (IIb) together constitute a completely general definition within ISM_D^L of being a *true shape* of L.

'TS X' abbreviates '($\mathrm{TS}_1\ X$ .v. $\mathrm{TS}_2\ X$)'.

Let us now consider some instances of these definitions. (I) is obvious enough, being so close to (I) of (V,A). Consider now some atomic universal sentence of L, say '$(x)x = x$'. The relevant instance of (IIa) is where the definiendum reads

'TS_1 (Lp Ex Rp Ex Id Ex)'.

This is defined as

'(Ea)((Lp Ex Rp Ex Id Ex) a . (Eb)(Ec)(Lp c . a C c,b . $(x)b$ Den x))'.

Hence the shape '$(x)x = x$' is a true shape if and only if there is at least one inscription having this shape which is true in the sense of (I).

Let us consider also a simple instance of (IIb). Let the relevant sentence of L be '$((x)x = x$ .v. $\sim(x)\mathrm{P}x$)'. This is a non-atomic universal sentence of L, and

'(Lp Lp Ex Rp Ex Id Ex Vee Tilde Lp Ex Rp Pee Ex Rp)'

is its structural description. Let this be abbreviated as 'A_0'. By (IIb), 'TS A_0' is then short for

'(Ea)(A_0 a . UnivSent a . $\sim$AtUnivSent a : TS_1 (Lp Ex Rp Ex Id Ex) .v. $\sim\mathrm{TS}_1$ (Lp Ex Rp Pee Ex))'.

A_0 is a true shape of L if and only if there is a non-atomic universal sentence of L which has this shape and either '$(x)x = x$' is a true shape or '$(x)\mathrm{P}x$' is not a true shape. And similarly, for more complicated instances.

The axioms of ISM_D^L comprise logical axioms analogous to those of (IV,D) but without axioms of identity, the syntactical rules *InSynR1a–InSynR17* above, translations of the non-logical axioms of L including axioms of identity, together with some semantical rules analogous to those of SM_D^L.

Let '$\boldsymbol{x}$', etc., stand arbitrarily for any translational variable of ISM_D^L. The first semantical rule required is then as follows.

InSemR1. $\vdash \boldsymbol{X}\ a\ .\supset.\ (\boldsymbol{x})(a\ \text{Den}\ \boldsymbol{x}\ .\equiv.\ ---\boldsymbol{x}---)$, if (i) '$---\boldsymbol{x}---$' is a sentential function of the one variable $\boldsymbol{x}$ from the translational part of ISM_D^L, and $\boldsymbol{X}$ is the structural description of an abstract '$(\boldsymbol{x})---\boldsymbol{x}---$', or (ii) $\boldsymbol{X}$ is 'Pee' and '$---\boldsymbol{x}---$' is '$\text{P}\boldsymbol{x}$'.

This rule stipulates, e.g., that

$$\vdash (\text{Ex Rp Ex Id Ex})\ a\ .\supset.\ (x)(a\ \text{Den}\ x\ .\equiv.\ x = x).$$

The second rule is essentially as in SM_D^L.

InSemR2. $\vdash a\ \text{Den}\ x\ .\supset.\ \text{PredCon}\ a.$

These are the only semantical rules required.

The consistency of ISM_D^L relative to that of IM and L jointly can very likely be shown by an adaptation of the argument of (V,G).

We cite now a few *syntactical* lemmas some of which will be needed below.

TF1a. $\vdash \boldsymbol{X}\ a\ .\ a\ \text{C}\ c,b\ .\ \text{Lp}\ c :\supset: \boldsymbol{X}'\ b$, if $\boldsymbol{X}$ is the structural description of an atomic universal sentence '$(\boldsymbol{x})---\boldsymbol{x}---$' from the translational part of ISM_D^L and $\boldsymbol{X}'$ is the structural description of the abstract '$(\boldsymbol{x})---\boldsymbol{x}---$'.

TF1b. $\vdash \boldsymbol{X}\ a\ .\supset.\ \text{AtUnivSent}\ a$, if (etc., about $\boldsymbol{X}$ as in *TF1a*).

TF2a. $\vdash \boldsymbol{X}\ a :\supset: (\text{E}b)\boldsymbol{X}_1\ b\ .\ (\text{E}b)\boldsymbol{X}_2\ b\ .\ \ldots\ .\ (\text{E}b)\boldsymbol{X}_n\ b$, if $\boldsymbol{X}$ is the structural description of a non-atomic universal sentence and $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ are the constituent atomic universal sentences contained in $\boldsymbol{X}$.

TF2b. $\vdash \boldsymbol{X}\ a :\supset: \text{UnivSent}\ a\ .\sim\text{AtUnivSent}\ a$, if (etc., about $\boldsymbol{X}$ as in *TF2a*).

TF2c. $\vdash \boldsymbol{X}\ a\ .\supset.\ \text{Sent}\ a$, if $\boldsymbol{X}$ is the structural description of a sentence.

TF3. $\vdash \boldsymbol{X}\ a\ .\ \text{Ex}\ b\ .\ \text{Rp}\ c\ .\ e\ \text{Like}\ a\ .\ d\ \text{C}\ b,c,e :\supset: \boldsymbol{X}'\ d$, if $\boldsymbol{X}$ is the structural description of a sentence '------' and $\boldsymbol{X}'$ is the structural description of a vacuous abstract '$x)$------'.

TF4a. $\vdash \text{Sent}\ a\ .\supset.\ (\text{E}b)(b\ \text{Part}\ a\ .\ \text{Rp}\ b)$.

TF4b. $\vdash \text{Sent}\ a\ .\supset.\ (\text{E}b)(b\ \text{Part}\ a\ .\ \text{Vbl}\ b)$.

G. Adequacy. What does it mean to say that the truth-predicate 'TS' is an *adequate* predicate? Here only slight changes need be made in the definition of adequacy given in (V,C). In particular we must insert a clause concerning the existence of inscriptions. Hence, we shall say that the predicate 'TS' is an *adequate* predicate for being a true shape in L if and only if every formula of ISM_D^L of the form

(T) $\qquad$ '$(\text{E}a)\boldsymbol{X}\ a :\supset: \text{TS}\ \boldsymbol{X}\ .\equiv.\ \boldsymbol{A}$'

is provable in ISM^L_D, where $\boldsymbol{A}$ is a sentence from the translational part of ISM^L_D and has $\boldsymbol{X}$ as its structural description.

To prove adequacy in this slightly modified sense, use must be made of course of the semantical axioms. The proof is in two parts corresponding to (IIa) and (IIb) of the definition. We first prove the case where $\boldsymbol{A}$ is an atomic universal sentence. And here we first prove that

(1) $\vdash (\mathrm{E}a)\boldsymbol{X}\,a\,.\,\mathrm{TS}_1\,\boldsymbol{X} :\supset: (x)\text{---}x\text{---},$

where $\boldsymbol{X}$ is the structural description of '(x)---x---', an atomic universal sentence from the translational part of ISM^L_D. Using *InSemR1* and the definition (IIa), we have that

(2) $\vdash \boldsymbol{X}\,a\,.\,\mathrm{AtUnivSent}\,a\,.\,\mathrm{Lp}\,c\,.\,a\,\mathrm{C}\,c,b\,.\,(x)b\,\mathrm{Den}\,x\,.\,\boldsymbol{X}'\,b :\supset: (x)(b\,\mathrm{Den}\,x\,.\equiv.\,\text{---}x\text{---}),$

if $\boldsymbol{X}$ is the structural description of '(x)---x---' and $\boldsymbol{X}'$ is the structural description of '$x)$---x---'. Hence, by *TF1a*,

$\vdash \boldsymbol{X}\,a\,.\,\mathrm{AtUnivSent}\,a\,.\,\mathrm{Lp}\,c\,.\,a\,\mathrm{C}\,c,b\,.\,(x)b\,\mathrm{Den}\,x :\supset: (x)\text{---}x\text{---},$

where (etc.). From this we immediately have (1). Now conversely,

$\vdash (x)(b\,\mathrm{Den}\,x\,.\equiv.\,\text{---}x\text{---})\,.\,(\mathrm{E}a)(\mathrm{E}c)(\boldsymbol{X}\,a\,.\,\mathrm{AtUnivSent}\,a\,.\,\mathrm{Lp}\,c\,.\,a\,\mathrm{C}\,c,b)\,.\,(x)\text{---}x\text{---}\,.\,\boldsymbol{X}'\,b :\supset: (\mathrm{E}a)(\mathrm{E}b)(\mathrm{E}c)(\boldsymbol{X}\,a\,.\,\mathrm{AtUnivSent}\,a\,.\,\mathrm{Lp}\,c\,.\,a\,\mathrm{C}\,c,b\,.\,(x)b\,\mathrm{Den}\,x),$

where (etc., as in (2)). We can drop the unneeded hypothesis by *InSemR1*, and use the definition (IIa). We then have that

$\vdash (\mathrm{E}a)(\mathrm{E}b)(\mathrm{E}c)(\boldsymbol{X}\,a\,.\,\mathrm{AtUnivSent}\,a\,.\,\mathrm{Lp}\,c\,.\,a\,\mathrm{C}\,c,b\,.\,\boldsymbol{X}'\,b)\,.\,(x)\text{---}x\text{---} :\supset: \mathrm{TS}_1\,\boldsymbol{X},$

where (etc., as in (2)). We eliminate the hypothesis

'$\boldsymbol{X}'\,b$'

by *TF1a*. Using *TE8a*, we have that

$\vdash (\mathrm{E}a)(\boldsymbol{X}\,a\,.\,\mathrm{AtUnivSent}\,a)\,.\,(x)\text{---}x\text{---} :\supset: \mathrm{TS}_1\,\boldsymbol{X},$

where (etc., about $\boldsymbol{X}$). But by *TF1b*, we can eliminate 'AtUnivSent a'. Hence we gain, using (1) also, that

(3) $\vdash (\mathrm{E}a)\boldsymbol{X}\,a :\supset: (x)\text{---}x\text{---}\,.\equiv.\,\mathrm{TS}_1\,\boldsymbol{X},$

where (etc.).

We now prove adequacy for (IIb), i.e., that

(4) $\vdash (\mathrm{E}a)\boldsymbol{X}\,a :\supset: \mathrm{TS}_2\,\boldsymbol{X}\,.\equiv.\,\boldsymbol{A}',$

where $\boldsymbol{X}$ is the structural description of $\boldsymbol{A}'$, and $\boldsymbol{A}'$ is a non-atomic universal sentence. We prove this by repeated use of (3). Let $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ be chosen as in the definition (IIb), i.e., as the constituent atomic universal sentences of $\boldsymbol{X}$. Let $\boldsymbol{X}_k$ ($1 \leqq k \leqq n$) be the struc-

tural description of A_k, and let A differ from A' only in containing '$\mathrm{TS}_1\ X_k$' wherever A' contains A_k'. By (3) we obviously have that

$$\vdash (\mathrm{E}a)X_1\ a :\supset: \mathrm{TS}_1\ X_1 .\equiv. A_1,$$
$$\vdash (\mathrm{E}a)X_2\ a :\supset: \mathrm{TS}_1\ X_2 .\equiv. A_2,$$

etc., and

$$\vdash (\mathrm{E}a)X_n\ a :\supset: \mathrm{TS}_1\ X_n .\equiv. A_n,$$

Combining these we have that

(5) $\vdash$ $(\mathrm{E}a)X_1\ a\ .\ (\mathrm{E}a)X_2\ a\ .\ \ldots\ .\ (\mathrm{E}a)X_n\ a\ .\ (\mathrm{E}b)(X\ b\ .\ \mathrm{UnivSent}\ b\ .\ \sim\mathrm{AtUnivSent}\ b) :\supset: (\mathrm{E}b)(X\ b\ .\ \mathrm{UnivSent}\ b\ .\ \sim\mathrm{AtUnivSent}\ b\ .\ A) .\equiv. A'$,

where (etc.). But from this we can get (4), simplifying the hypothesis by *TF2a* and *TF2b*.

Combining (3) and (4), we gain the

Adequacy Rule 1. $\vdash (\mathrm{E}a)X\ a :\supset: \mathrm{TS}\ X .\equiv. A$, if X is the structural description of A and A is a sentence from the translational part of ISM_D^L.

H. Truth and Vacuous Abstracts. Another way of defining the semantical truth-concept for L suggests itself if we use vacuous abstracts, as in (V,B). By this method we shall be able to define a truth-predicate applying to inscriptions and not merely to shapes and yet not encountering the difficulty concerning inscriptional existence.

The full definition will be given in several steps. First, for sentences a which are not too long, we can say that a is true in L if and only if a vacuous abstract d, consisting of a variable concatenated with a Rp concatenated with a, denotes every object.

(I) '$\mathrm{Tr}_1\ a$' abbreviates '(Sent a . (Eb)(Ec)(Ed)(Vbl b . Rp c . d C b,c,a . $(x)d$ Den x))'.

This definition introduces a truth-concept applying to all but a few sentences of L. But suppose a were itself so long as to exhaust all of space-time, i.e., to contain all the characters of L available. Then the d required in the definiens would not be forthcoming, for d is longer than a by 2 typographical characters. If a is a sentence exhausting the whole of space-time, or exhausting it to within one character, e.g., its truth-concept cannot be expressed by 'Tr_1'. Let

'VacAbst a' abbreviate '(Sent a . (Eb)(Ec)(Ed)(Vbl b . Rp c . d C b,c,a))'.

We say that a sentence a has a *vacuous abstract,* if and only if there is an abstract d which consists of a variable followed by an isolated Rp followed by a. 'Tr_1' applies only to sentences which have a vacuous abstract in this sense.

This limitation may be remedied by supplementing the definition of 'Tr_1' as follows. If a sentence *a* has a vacuous abstract we shall say it has the form (1). If *a* is of such great length as *not* to have a vacuous abstract, it will have the form (2) of an atomic universal sentence, or (3) of a disjunction, or (4) of a formula beginning with a string of Tildes. Clearly all sentences of *L* will be of one of these four forms. (Cf. *TE8b*.) In case (2), we let

'Tr_2 *a*' abbreviate '(~VacAbst *a* . AtUnivSent *a* . (E*b*)(E*c*)(*a* C *b*,*c* . Lp *b* . (*x*)*c* Den *x*))'.

This case is similar to (I) of § F. In the case (3) of a disjunction, we let

'Tr_3 *a*' abbreviate '(Sent *a* . ~VacAbst *a* . (E*b*)(E*c*)(*a* D_s *b*,*c* : Tr_1 *b* .v. Tr_1 *c*))'.

Here a sentence *a* is Tr_3 if and only if *b* is Tr_1 or *c* is Tr_1, *a* being a strict disjunction of *b* with *c*. Note that 'Tr_1' is applicable to *b* and *c*, because such inscriptions are sentences (*TE8c*) and hence contain at least one Rp (*TF4a*) and at least one Vbl (*TF4b*). Hence *c* has a vacuous abstract, because the extra Vbl and Rp are always forthcoming from *b*. And similarly for *b*, because the extra Vbl and Rp are always forthcoming from *c*.

Under case (4) we have two possibilities, depending upon whether the string of Tildes involved is odd or even. Here we shall need two truth-definitions. For these we need some preliminary notions. An inscription *a* is an *even string of typographical characters* if and only if it is the concatenate of two equally long inscriptions.

'EvStr *a*' abbreviates '(E*b*)(E*c*)(*b* EqLng *c* . *a* C *b*,*c*)'.

An inscription *a* is a *string of Tildes* if and only if every character which is a segment of it is a Tilde.

'TildeStr *a*' abbreviates '(*b*)(Char *b* . *b* Seg *a* :⊃: Tilde *b*)'.

And then an inscription is obviously an *even string of Tildes* if and only if it is an even string and is a string of Tildes.

'EvTildeStr *a*' abbreviates '(TildeStr *a* . EvStr *a*)'.

Similarly

'OddTildeStr *a*' abbreviates '(TildeStr *a* . ~EvStr *a*)'.

(Cf. the corresponding definitions in (VII,E).) In the case (4) of an even string of Tildes we have the following truth-definition.

'Tr_4 *a*' abbreviates '(Sent *a* . ~VacAbst *a* . (E*b*)(E*c*)(EvTildeStr *b* . *a* C *b*,*c* . ~(E*d*)(E*e*)(Tilde *d* . *c* C *d*,*e*) . (Tr_2 *c* .v. Tr_3 *c*)))'.

A sentence consisting of an even string of Tildes followed by a sentence *c* not beginning with a Tilde is true if and only if *c* is true.

But c here will be either an atomic universal sentence or a disjunction and hence either 'Tr_2' or Tr_3' is applicable to it.

In the case (4) of an odd string of Tildes, we let

'$\mathrm{Tr}_5\ a$' abbreviate '(Sent a . $\sim$VacAbst a . (Eb)(Ec)(OddTildeStr b . a C b,c . $\sim$(Ed)(Ee)(Tilde d . c C d,e) . $\sim$($\mathrm{Tr}_2\ c$ .v. $\mathrm{Tr}_3\ c$)))'.

A completely general definition of truth can now be given by disjunction.

'Tr a' abbreviates '($\mathrm{Tr}_1\ a$ .v. $\mathrm{Tr}_2\ a$ .v. $\mathrm{Tr}_3\ a$ .v. $\mathrm{Tr}_4\ a$ .v. $\mathrm{Tr}_5\ a$)'.

This truth-predicate will be said to be *adequate* if and only if every formula of the form

$$\text{'}\boldsymbol{X}\ a :\supset: \mathrm{Tr}\ a .\equiv. \boldsymbol{A}\text{'}$$

is provable in ISM_D^L, where $\boldsymbol{A}$ is a sentence from the translational part of ISM_D^L and has $\boldsymbol{X}$ as its structural description.

The proof of adequacy for 'Tr' in this sense is omitted. The statement of this law is the *Adequacy Rule 2*. In view of this law and the *Adequacy Rule 1* we have the following equivalence law.

TH1. $\vdash$ Sent a . $\boldsymbol{X}\ a :\supset:$ TS $\boldsymbol{X}$.$\equiv$. Tr a, for any $\boldsymbol{X}$.

On the basis of either of the Adequacy Rules, using the Rule of Shapes, we can prove that the object-language L is consistent, by an adaptation of the argument of (V,E).

Consistency Rule. $\vdash$ (a)(b)(Sent a . b N $a :\supset: \sim$(Thm a . Thm b)).

Various further notions of semantics definable in terms of the truth-concept may now be introduced and their fundamental properties proved, as in (V,F). Also one can develop in ISM_D^L an extensive theory of predicate constants, as in (IV,F).

Also with only slight changes, L may be taken as a language containing one or more primitive individual or functional constants. An inscriptional semantics for L can then be given along the lines of Chapter X. (See (XII,E) below.)

CHAPTER XII

INSCRIPTIONS AND COMPREHENSION

In the preceding chapter an inscriptional semantics was formulated in which a relation of multiple denotation was taken as a semantical primitive. In the present chapter we shall be concerned with an inscriptional semantics based on a relation of comprehension. The semantics of the preceding chapter was translational in character, containing *in toto* a translation of the object-language L. The semantics of the present chapter is non-translational, and is hence akin to the semantical meta-language SM_C^L. But the values for expressional variables here are inscriptions, and the main problem therefore will be to reconstruct non-translational semantics in such a way as not to assume an infinity of expressions.

In § A the semantical preliminaries are presented, including some *basic definitions* and the *Semantical Rules*. § B is devoted to noting some *elementary theorems*. In § C the *truth-concept* and some derivative notions of semantics are defined. In § D the *consistency* of the object-language is proved. A theory of *individual and functional constants* within inscriptional semantics is sketched in § E. A *formal theory of quotation marks* is developed in § F within inscriptional semantics of a translational kind. Finally, § G is devoted to some general comments concerning semantical meta-languages containing both inscriptions and shapes as values for variables.

A. Semantical Preliminaries. The semantical primitive here is a relation of comprehension taking inscriptions of a given object-language L as relata. Let 'a', 'b', etc., be the inscriptional variables, as in the preceding chapter, and let 'a Cmprh b' symbolize that the inscription a comprehends the inscription b. The meaning assigned to 'Cmprh' is similar to that of Chapter VIII, so that 'a Cmprh b' is to hold just where a denotes everything which b does. Let the semantical meta-language here be called ISM_C^L, the inscriptional semantical meta-language for L based on comprehension.

The syntactical part of ISM_C^L is described in §§ A–E of the preceding chapter. The object-languages likewise are of the kind L considered there. Here the object-language does not reappear translated into ISM_C^L, so that we need only one style of variables, the expres-

sional variables. The structure of ISM^L_C is thus considerably simpler than that of ISM^L_D.

Several semantical notions are definable here essentially as in (VIII,B), such as 'DE *a*', '*a* CoExt *b*', etc.

The logical and syntactical axioms of ISM^L_C are the logical and syntactical axioms of ISM^L_D. The semantical rules are essentially as in (VIII,D), but with some slight changes due to the different treatment of concatenation. For convenience, we list the semantical rules here in full, even though some are typographically precisely as in (VIII,D).

InSemR1. ⊢ DE *a* .≡. PredCon *a*,

InSemR2. ⊢ *a* Cmprh *b* . *c* Cmprh *a* :⊃: *c* Cmprh *b*,

InSemR3. ⊢ (E*d*)(E*e*)(E*d*′)(E*e*′)(SentFuncOne *d*,*e* . SentFuncOne *d*′,*e*′ . *e* Like *e*′ . (E*a*′)(E*b*′)(Rp *a*′ . *b*′ Like *e* . *b* C *b*′,*a*′,*d*) . (E*a*′)(E*b*′)(Rp *a*′ . *b*′ Like *e* . *c* C *b*′,*a*′,*d*′) . (E*a*′)(E*b*′)(E*c*′)(Rp *a*′ . *b*′ Like *e* . *c*′ D_s *d*,*d*′ . *a* C *b*′,*a*′,*c*′)) .⊃. *a* Sum *b*,*c*,

InSemR4. ⊢ Hyps (of *InSemR3*, but with 'C_s' in place of 'D_s') .⊃. *a* Prod *b*,*c*,

InSemR5. ⊢ (E*c*)(E*d*)(E*e*)(SentFuncOne *c*,*d* . (E*b*′)(E*d*′)(*d*′ Like *d* . Rp *b*′ . *a* C *d*′,*b*′,*c*) . *e* N *c* . (E*b*′)(E*d*′)(*d*′ Like *d* . Rp *b*′ . *b* C *d*′,*b*′,*e*)) .⊃. *b* Neg *a*,

InSemR6. ⊢ *d* Neg *b* . *c* Prod *a*,*d* . ∼*b* Cmprh *a* :⊃: ∼Null *c*,

InSemR7. ⊢ *d* Sum *a*,*b* . *d* Cmprh *c* . (*e*)(*c* Cmprh *e* . (*a* Cmprh *e* .v. *b* Cmprh *e*) :⊃: Null *e*) :⊃: Null *c*,

InSemR8. ⊢ (E*b*)(*a* Clsr′ *b* . LogAx *b*) . (E*b*)(E*d*)(Vbl *b* . Rp *d* . *c* C *b*,*d*,*a*) :⊃: Univ *c*,

InSemR9. ⊢ Univ *a* . Null *b* :⊃: ∼*b* Cmprh *a*,

InSemR10. ⊢ SentFuncOne *b*,*a* . Vbl *a*″ . (E*e*)(E*a*′)(Rp *e* . *a*′ Like *a* . *c* C *a*′,*e*,*b*) . Univ *c* . (E*e*)(E*a*′)(E*b*′)(E*c*′)(Rp *e* . *a*′ Like *a* . Lp *c*′ . Rp *b*′ . *d* C *a*′,*e*,*c*′,*a*″,*b*′,*b*) :⊃: Univ *d*,

InSemR11. ⊢ Sent *a* . Vbl *b* . Vbl *c* . (E*c*′)(Rp *c*′ . *d* C *b*,*c*′,*a*) . (E*b*′)(Rp *b*′ . *e* C *c*,*b*′,*a*) :⊃: *d* Cmprh *e*,

InSemR12. ⊢ Sent *a* . Vbl *b* . (E*e*)(Rp *e* . *c* C *b*,*e*,*a*) :⊃: Univ *c* .v. Null *c*,

InSemR13. ⊢ SentFuncOne *a*,*b* . *b* FV *a* . *d* SF^c_b *a* . (E*b*′)(E*c*′)(*b*′ Like *b* . Rp *c*′ . *e* C *b*′,*c*′,*a*) . (E*b*′)(E*c*′)(*b*′ Like *c* . Rp *c*′ . *e*′ C *b*′,*c*′,*d*) :⊃: *e* Cmprh *e*′,

and

InSemR14. ⊢ Pee *a* . Rp *b* . Vbl *c* . *c*′ Like *c* . Pee *d* . *e* C *c*,*b*,*d*,*c*′ :⊃: *a* CoExt *e*.

These rules are closely patterned after the General Rules of Comprehension of SM^L_C and play within the meta-language here an

analogous role. Note, however, the additional clauses concerning inscriptional likeness. The reason these are needed is that a character cannot be repeated within an inscription (*InSynR13* in (XI,A)). Rather than repeat a character a one must use another character like a.

Note also that *InSemR11* and *InSemR13* differ from *GRC11* and *GRC13* respectively in not containing clauses of the form '$\sim b = c$'. Nor do we have here clauses of the form '$\sim b$ Like c' in their place. The reason for this is to assure the following theorem.

TA1. $\vdash$ PredCon a . a Like b :$\supset$: a Cmprh b.

And hence also

TA2. $\vdash$ a Like b . a Cmprh c :$\supset$: b Cmprh c,

and

TA3. $\vdash$ a Like b . c Cmprh a :$\supset$: c Cmprh b.

The analogues of these, with '=' in place of 'Like', of course hold here as well as in SM_C^L. If '$\sim b$ Like c' were taken as additional hypotheses in *InSemR11 and InSemR13*, we should wish to include *TA1* or alternatively, *TA2* together with *TA3*, as additional rules.

B. Some Theorems. On the basis of the semantical rules *InSemR1–InSemR14*, one can develop a general theory of comprehension within ISM_C^L along the lines of Chapter IX. One can prove various theorems concerning comprehension, co-extensiveness, semantical sums, products, and negatives, and common, null, and universal predicate constants. Here one often needs additional clauses concerning the likeness of inscriptions, such as those occurring in the inscriptional semantical rules. Bearing this slight change in mind, practically all the theorems of Chapter IX, §§ A–D, can be reformulated appropriately and proved within ISM_C^L.

C. Truth and Derivative Notions. Of the two ways of defining a semantical truth-concept within the inscriptional semantics of Chapter XI, only one of them is available here. The first definition given, in (XI,F), makes fundamental use of notions from the translational part of the meta-language, and thus can be given only within the framework of a translational semantics. The second definition of Chapter XI, in § H, makes no use of the translations of expressions of L, and hence may be adapted here.

Within the non-translational semantics of this chapter as within that of Chapters VIII–IX, the requirement that the truth-concept defined be adequate must, in a certain sense, be abandoned. But, as

we have seen, this does not mean that adequacy is in any way denied here or its negative assumed. It is rather that the requirement cannot be stated for ISM^L_C, because this meta-language does not contain translations of the expressions of L. The dropping of this requirement is much less important than one might have supposed, bearing in mind the vital role which adequacy plays within translational semantics. (See below, (XIII,F).)

Reasons for philosophical interest in so restricted a meta-language as ISM^L_C will be given in the next chapter in some detail.

D. Consistency and Relative Consistency. If we take L for the moment as S we can carry out the consistency proof of S within ISM^L_C. The details are similar to those of (IX,F). Rather than to assume outright that certain predicate constants of S are universal, we should here as in Chapter IX take such assumptions as hypotheses wherever needed. The Specific Hypotheses of Comprehension needed for S are similar to those of (IX,E).

Also the consistency of ISM^L_C relative to that of the underlying inscriptional syntax can very likely be shown by an appropriate adaptation of the argument of (IX,G).

E. Individual Constants and Inscriptions. Suppose that L contains a finite number of individual constants as primitive. Bearing in mind the material of Chapter X, only slight changes need be made in the semantical meta-language, whether it be translational or non-translational. First, let us consider again ISM^L_D.

Let 'Ay' be the additional primitive structural-descriptive predicate. 'Ay a' can be read 'a is an *ay*-shaped character'. And let '$\bar{a}$', '$\bar{a}''$', etc., be the translations within ISM^L_D of the individual constants of L. We can define

'Ay′ a' as '(Eb)(Ec)(Ay b . Ac c . a C b,c)',
'Ay″ a' as '(Eb)(Ec)(Ay′ b . Ac c . a C b,c)',

and so on, until we have a structural-descriptive predicate for every individual constant of L. The definition of 'InCon a' can then be given by enumeration.

We let '$\boldsymbol{a}$' with or without numerical subscripts stand for any translational individual constant of ISM^L_D, and '***Ay***' for 'Ay', 'Ay″', 'Ay‴', or etc. Also we may say that 'Ay' is the *structural description* of '$\bar{a}$', '(Ay⌒Ac)' is the structural description of '$\bar{a}''$', and so on.

The relation of being a *proper name of* is definable within ISM^L_D essentially as in SM^L_D (X,B).

'a PrNm x' abbreviates '(InCon a . (Eb)(Ec)(Ed)(Ee)(Ea') (Vbl b . Rp c . d Like b . Id e . a' C b,c,d,e,a) . a' Den x)'.

The following three basic theorems concerning individual constants are then forthcoming.

TE1. $\vdash a$ PrNm x .⊃. InCon a.

TE2. $\vdash$ ***Ay*** a .⊃. a PrNm ***a***, if ***Ay*** is the structural description of ***a***.

TE3. $\vdash$ InCon a . a PrNm x . a PrNm y :⊃: $x = y$.

The handling of individual constants within ISM_C^L is similar, utilizing the material of (X,D). Here, as there, a relation of being the proper name of cannot be introduced, because a non-translational meta-language lacks the appropriate expressions. But important properties of individual constants can be proved nonetheless as in (X,E).

If one or more functional constants are introduced in L as primitives, ISM_D^L and ISM_C^L need be altered only slightly along the lines of (X,F).

F. Formal Theory of Quotation Marks. Because of the widespread interest in the distinction between the use and mention of expressions, the importance of the proper use of quotation marks in methodological writing is now beyond question. Precise rules for the use of quotation marks have rarely if ever been formulated. The principles concerning them are usually left at a vague, intuitive level. Some interest, therefore, might attach to formal definitions of various uses of quotation marks within the framework of a carefully formulated semantical meta-language. We shall formulate such definitions in this section within the framework of ISM_D^L. We do this in ISM_D^L rather than within SM_D^L in order to define an inscriptional quote function as well. Some theorems governing these quote functions will be given. L is taken here as containing one or more primitive individual constants but no primitive functional constants.

Single quotation marks will be introduced in the customary sense, so that the result of applying single quotes to a symbol will be construed as a shape. Within ISM_D^L, shapes figure as properties, *shape properties* as we have called them above. Suppose ' '(' ' is defined as 'Lp'. Then, in place of writing

'Lp a',

we could write instead

' '(' a'.

Because expressions standing for shapes occur primitively within ISM_D^L only as one-place predicate constants, so also will expressions

containing quotation marks. But just as above in (XI,A) we introduced expressions such as

$$`(S_1 \ldots S_n)\ a',$$

so also we can define single quotation marks as enclosing a long sequence of expressions.

Let 'S_1', 'S_2', etc., be used as in (XI,A) to stand for any primitive structural-descriptive predicates of ISM_D^L. And in a similar way let 's_1', 's_2', etc., stand for any of the translational symbols of ISM_D^L,

$$`(', ')', `\sim', `\vee', `P', `\bar{a}', `x', `=', \text{ or } ` \ ' .$$

's_1', etc., are then another kind of syntactical variable that will be useful in the meta-language of ISM_D^L. We can now define

' '$s_1 s_2 \ldots s_n$' a' as '$(S_1 S_2 \ldots S_n)\ a$', if S_i is the structural description of s_i $(1 \leq i \leq n)$.

According to this definition, e.g.,

$$` `(x)x = x'\ a'$$

states that the inscription a has the shape consisting of an Lp followed by an Ex followed by an Rp followed by an Ex followed by an Id and finally followed by an Ex.

This definition introduces formally single quotation marks within ISM_D^L. Strictly the usefulness of such a definition is somewhat limited. Shapes in ISM_D^L are not values for variables, and expressions for them occur only as predicates. Nor, in virtue of $TE1$ are shapes proper names. Only *inscriptions* of L denote or can be proper names and they must be either predicate or individual constants of L.

But nonetheless the formal definition of single quotation marks given is not a mere idle curiosity. We can imagine a user of ISM_D^L, trained to it from childhood, who would find this definition of interest. Suppose he had never used quotation marks. All the informal comment about ISM_D^L given above in its meta-language, our imagined user had absorbed from childhood, so that he came to know ISM_D^L much as children come to know their native tongue. He might have observed single quotation marks in logical treatises or elsewhere and been puzzled by their meaning. The definitional schema given supplies an appropriate meaning, and for him would be genuinely informative.

It would be idle to object to this definition on the grounds that a great bevy of quotation marks is needed to set it up, so that to make sense of it we must already understand how quotation marks are used. To set up any language-system carefully we may use quotation marks *within the meta-language*, and if we do so we must there be able to handle them correctly. But this is a very different matter from

giving a formal definition of quotation marks as they occur in a semantical meta-language taken as object-language.

A more interesting kind of quotation function is of an inscriptional kind. Inscriptions being values for variables within ISM_D^L, we shall find some interesting uses for it. The definition presupposes the Hilbert ε-function (II,F). Let

'$\cdot s_1 \ldots s_n \cdot$' abbreviate '$(\varepsilon a.(S_1 S_2 \ldots S_n)\ a)$', where S_i is the structural description of s_i $(1 \leq i \leq n)$, and $n \geq 2$,

and

'$\cdot s_1 \cdot$' abbreviate '$(\varepsilon a.S_1\ a)$', if S_1 is the structural description of s_1.

According to this definition,

$$\cdot(x)x = x\cdot,$$

e.g., is any inscription having the shape

$$'(x)x = x'.$$

But ε-expressions are defined only in context, so that

$$'\cdot(x)x = x\cdot'$$

is defined only contextually. Hence, according to the definition

$$\cdot(x)x = x\cdot$$

can be said to have a given property Q if and only if there is at least one inscription of the shape '$(x)x = x$', and every inscription which has this shape has the property Q.

Frequently in inscription theory we wish to speak of any inscription having such and such a shape, and to ascribe to *any* such inscription some general property. The inscriptional quote function just defined facilitates such statements. E.g., to say that any Ay is a proper name of the individual ā is to say that

$$(\varepsilon a.\text{Ay}\ a)\ \text{PrNm}\ \bar{a},$$

i.e., that

$$\cdot\bar{a}\cdot\ \text{PrNm}\ \bar{a}.$$

The following elementary theorems concerning this inscriptional quote function are easy to prove within ISM_D^L. (The scope of an ε-expression is to be taken as the shortest exhibited sentential context of ISM_D^L containing that expression. In the case of formulae containing two or more ε-expressions, the scope of the left one is to be the longer.)

TF1. $\vdash \cdot\boldsymbol{a}\cdot\ \text{PrNm}\ \boldsymbol{a}$.

By *TE2* and the existence laws that $\vdash (\text{E}a)\boldsymbol{Ay}\ a$, where $\boldsymbol{Ay}$ is the structural description of $\boldsymbol{a}$.

TF2. $\vdash \cdot \boldsymbol{a} \cdot$ PrNm $\boldsymbol{a}_1 . \equiv . \cdot \boldsymbol{a}_1 \cdot$ PrNm $\boldsymbol{a}$.

By *TE2*, *TE3*, and existence laws.

TF3. $\vdash \cdot \boldsymbol{a} \cdot$ PrNm $\boldsymbol{a}_1 . \cdot \boldsymbol{a}_1 \cdot$ PrNm $\boldsymbol{a}_2 : \supset : \cdot \boldsymbol{a} \cdot$ PrNm $\boldsymbol{a}_2$.
TF4. $\vdash \cdot \boldsymbol{a} \cdot$ Like $\cdot \boldsymbol{a} \cdot$.

(Note, however, that

$$`\cdot \boldsymbol{a} \cdot = \cdot \boldsymbol{a} \cdot'$$

does not hold.)

TF5. $\vdash \cdot \boldsymbol{a} \cdot$ Like $\cdot \boldsymbol{a}_1 \cdot . \equiv . \cdot \boldsymbol{a}_1 \cdot$ Like $\cdot \boldsymbol{a} \cdot$.
TF6. $\vdash \cdot \boldsymbol{a} \cdot$ Like $\cdot \boldsymbol{a}_1 \cdot . \cdot \boldsymbol{a}_1 \cdot$ Like $\cdot \boldsymbol{a}_2 \cdot : \supset : \cdot \boldsymbol{a} \cdot$ Like $\cdot \boldsymbol{a}_2 \cdot$.
TF7. $\vdash \cdot \boldsymbol{a} \cdot$ Like $\cdot \boldsymbol{a}_1 \cdot . \supset . \boldsymbol{a} = \boldsymbol{a}_1$.

(The converses of theorems stipulated by *TF7* clearly do not hold unless distinct primitive individual constants are taken as denoting distinct individuals.)

TF8. $\vdash \sim(\mathrm{E}x) \cdot \boldsymbol{x} \cdot$ PrNm x.

(Variables are not proper names.)

Let '$\boldsymbol{P}$', with or without numerical subscripts, stand for any one-place predicate constant in the translational part of ISM_D^L of length $\leq k_0$ (k_0 as in (XII,A)). We have then the following theorems.

TF9. $\vdash \cdot \boldsymbol{P} \cdot$ Ext $\boldsymbol{x} \ni \boldsymbol{P}\boldsymbol{x}$.
TF10. $\vdash \cdot \boldsymbol{P} \cdot$ Ext $\boldsymbol{x} \ni \boldsymbol{P}_1 \boldsymbol{x} . \equiv . \cdot \boldsymbol{P}_1 \cdot$ Ext $\boldsymbol{x} \ni \boldsymbol{P}\boldsymbol{x}$.
TF11. $\vdash \cdot \boldsymbol{P} \cdot$ Like $\cdot \boldsymbol{P} \cdot$.
TF12. $\vdash \sim(\mathrm{E}x) \cdot \boldsymbol{A} \cdot$ Den x, if $\boldsymbol{A}$ is a formula from the translational part of the meta-language.

(Formulae do not denote.)

Note that all of these quotation functions involve expressions which are translations of expressions of L. Hence these functions are introduced only within a translational meta-language. Within a non-translational meta-language such expressions are not available. We simply lack there, so to speak, the expressions to put quotes around.

G. Semantics Based upon Inscriptions and Shapes. For some purposes one may wish to employ a semantical meta-language in which both inscriptions and shapes are taken as values for variables. We shall not sketch here such a meta-language in any detail, but offer only a few comments.

The most natural procedure in constructing such a meta-language would no doubt be to regard shapes as classes of similar inscriptions and hence as values for class variables of type one higher than that of variables for inscriptions. Within such a language, class variables would be needed fundamentally. If one adopts the view taken above,

one would not wish to assume that there is an infinite number of inscriptions. Even this kind of a semantical meta-language, if non-translational, could be formulated as a first-order language in the narrow sense. But if it is translational, it could be of first order in the narrow sense if and only if the object-language concerned is also.

The primitives of such a meta-language would presumably include something akin to a concatenation operation. Let 'Lp', etc., designate primitively the appropriate shapes or classes of similar inscriptions. Should concatenation be applied primitively to shapes or to inscriptions? It seems more natural to apply it to inscriptions and a concatenation operation applying to shapes would then presumably be definable. E.g., '(Lp⌒Rp)' is definable as 'the class of all inscriptions *a* such that there are inscriptions *b* and *c* such that *b* is an Lp, *c* is an Rp and *a* is the inscriptional concatenate of *b* and *c*'. If there are no such inscriptions the concatenate of the two shapes concerned is the null class of inscriptions.

For the purposes of a denotational semantics of the kind discussed in this book, a meta-language containing both inscriptions and shapes is not needed. But such a meta-language would undoubtedly be useful for certain more general purposes, e.g., in studying the interrelations of inscriptions and shapes and of the different ways in which they function within discourse. Also for the purposes of a full semiotical description of a language-system of wide expressive power and containing exact correlates of the egocentric particulars (words such as 'I', 'now', 'here', 'this', etc.), a meta-language containing both inscriptions and shapes as values for variables would very likely be needed.

CHAPTER XIII

FIRST-ORDER CONSTRUCTIVISM

THE general methodological maxim, stated in (II,H), that denumerable, first-order languages are preferable to other kinds of systems, has been explicitly followed in the formulation of the various semantical meta-languages above. The reasons for accepting this maxim have not been indicated. The preference for such languages might therefore be thought to be somewhat arbitrary and the reasons for it vague or otherwise ill-founded. But, on the contrary, there are several important arguments in favor of narrow, first-order systems. Some of these are logico-mathematical in character, some are somewhat more philosophical, some are based to some extent upon matters of preference. These various arguments will be put forward in the present chapter in some detail. Many of them will perforce be somewhat rough and in need of more careful formulation. But they will perhaps help to clarify the broad philosophical position underlying the general methodological maxim, and hence underlying the various semantical meta-languages formulated in this book.

The view to be put forward we have called *first-order constructivism.*

In § A the *purport* of the general methodological maxim is indicated in fuller detail than heretofore. In § B *arguments in behalf of* this maxim are given. In § C and § D the general point of view of first-order constructivism is compared and contrasted with *nominalist* and *realist* views. In § E arguments are given *in behalf of non-translational semantics.* In § F the various formulations of non-translational semantics are shown to be *minimal* in an appropriate sense of that word. A possible objection to non-translational semantics is given in § G together with a defense. In § H two general theories of semantical systems are sketched in which semantical systems appear either as *values for variables* or as *virtual ordered couples of virtual relations.* Finally in § I another argument in behalf of the general methodological maxim is given in connection with the "situation" in the foundations of mathematics known as the *Skolem paradox.*

A. The Purport of the Maxim. The general methodological principle to prefer wherever possible first-order languages in the narrow

sense is concerned with *all* first-order language-systems whatsoever, whether as object-languages or as syntactical or semantical meta-languages. In the case of object-languages, the greatest freedom of choice has been allowed throughout the preceding chapters. There have been no restrictions whatsoever upon them other than that they be of first order, and the semantical methods developed have been applicable, with a few exceptions, to all first-order object-languages uniformly. Thus the general methodological maxim has not been followed in the choice of object-languages. The reason for this is that methods are not known for formulating (the whole of) mathematical analysis (the theory of real numbers), and hence areas of mathematics or philosophy presupposing it, using only first-order object-languages with a denumerable fundamental domain. Nor is it clear that one can use only denumerable first-order object-languages for the purposes of science. Although we may hope to show that such object-languages will suffice for most purposes, this can by no means by substantiated on the basis of present knowledge. The weight of evidence seems to be on the other side. Science and mathematics are shot through and through with concepts resting fundamentally upon the non-denumerable. The most important known ways of defining such concepts are within formalisms of higher order or of first order only in the wider sense. In the absence, then, of methods of formulating various areas of science and mathematics within denumerable first-order systems, we must employ more powerful object-languages.

We have already noted in Chapter III how, using the work of Tarski, Chwistek, and Quine, a denumerable, first-order *syntax* can be formulated, so that the use of the general maxim leads to nothing new in syntax.

In the case of *semantical* meta-languages, the situation has been quite different. All semantical meta-languages (of object-languages of sufficient power) which have heretofore been formulated, if of first order at all, have been of first order only in the wider sense. The use of the general methodological maxim here might therefore be thought too restrictive. We have shown, however, that this is not the case. The translational semantical meta-language SM_D^L is of first order in the narrow sense provided L is. If L is of first order only in the wider sense, as in the case of systems based upon type theory or set theory, then of course SM_D^L is a first-order language in the wider sense, even though only restricted syntactical and semantical notions are used. In SM_C^L, as we have seen, we have a denumberable, first-order semantical meta-language, irrespective of L. The use of the general maxim has thus guided us in gaining a very simple formulation of

translational semantics. And it has also led to an altogether new semantics of the non-translational kind.

In having achieved a completely general, denumerable, non-translational, first-order semantics, we might think that there is then no further need of seeking denumerable, first-order *object*-languages of greater and greater power. We can provide a full syntactical and semantical description of any first-order object-language of any complexity within a non-translational semantics. At the meta-linguistic level our ideal has been completely realized. We can handle the syntax of a language L, determine whether a given expression of L is a formula or a sentence or not, and show what formulae or sentences follow from it by means of the rules of inference. More than this, we know what it means to say that sentences of L are true and hence we gain for the language an explicit interpretation. For this, use is made of Specific Hypotheses of Comprehension.

Nonetheless the general methodological maxim is intended to apply to all language-systems whatever, including object-languages. The arguments to be given in favor of the maxim are also arguments in favor of denumerable, first-order object-languages. When we have formulated such an object-language, adequate for a given area of scientific, philosophical, or mathematical theory, the semantical meta-languages of that language are then simpler than for more complicated systems. This is obvious in the case of translational meta-languages, because they contain their object-languages as a part. But also, in a certain sense, this holds of non-translational meta-languages as well. (See § G below.)

B. Consistency and Economy. The importance of the concept of consistency has already been indicated to some extent above (V,E). Obviously language-systems which can be proved consistent in convincing ways are preferable to language-systems known to be contradictory. A language-system known to be contradictory is rarely of interest unless the inconsistency can easily be remedied. More often we are confronted with language-systems, either as object- or as meta-languages, which are not known to be either consistent or inconsistent, or which if known to be consistent are such only by means of an argument or proof so complicated or using such powerful logico-mathematical devices as scarcely to carry conviction.

An important reason for preferring denumerable, first-order language-systems to languages of other forms, is that many such systems may be proved consistent, using only cogent and convincing methods. Most languages for which such consistency proofs exist are in fact of this kind. Having first-order semantical meta-languages before us,

for which very simple relative consistency proofs are available, we are assured of secure foundations for semantics. Heretofore semantical meta-languages have ordinarily been of higher order or of first order only in the wider sense. For such languages the consistency or relative consistency proofs available are usually of a more complicated kind.

Another reason for preferring first-order language-systems, as over and against language-systems of higher order, is in connection with economy. Rather than a multiplicity of different styles of variables of different logical types, we have in first-order systems only variables of a single type. Rather than primitive predicates applying ambiguously to entities of different logical types, we need only primitive predicates applying exclusively to individuals. Also there is economy in the axiomatic framework. In place of logical axioms and rules covering different styles of variables and quantifiers, we need only the simple rules of first-order logic.

But there is also ontological economy, i.e., economy in the number of objects. Given a denumerable, first-order L, the number of objects which can be values for variables within L is of course denumerable. Given a second-order L with axioms or rules sufficient to show that there is at least a denumerable number of individuals in L, one can show that the number of classes which are values for variables is non-denumerable. Within the fundamental domains of such languages there are thus vastly more objects than within the fundamental domains of narrow, first-order L's.[1]

C. Semantic Constructivism, Nominalism, Realism, Finitism. We have called the philosophical point of view underlying the general methodological maxim *first-order constructivism*. Use of the maxim as applied exclusively to semantical meta-languages might be called *semantic constructivism*. In this section various comments concerning first-order and semantic constructivism are put forward, as well as a comparison of this view with the well-known alternative views of nominalism and realism.

The word 'constructivism' is used by mathematicians and logicians in various different senses. One especially influential use of this word is in connection with the intuitionistic views of Brouwer in the foundations of mathematics. But here the point of view is classical

[1] The use of 'economy' here is somewhat vague. Economy and simplicity are no doubt at some point interrelated, and important work has been done by Goodman and others in clarifying the notion of simplicity. See N. Goodman, *The Structure of Appearance*, pp. 59–75. Also J. Kemeny, "The Use of Simplicity in Induction", *Philosophical Review*, 62 (1953), pp. 391–408.

as opposed to intuitionistic, the laws of the classical, two-valued theory of truth-functions and of quantifiers being presupposed *in toto*. There are other influential uses of the word also. But there seems to be no generally accepted *single* usage of the word, so that its use here in a philosophical context does no violence to accepted mathematical usage. Also, when we speak of "first-order constructivism", the qualifying phrase indicates sufficiently what is meant. The view might be called more simply 'first-order-ism', or 'first-order denumerabilism', but these phrases are scarcely palatable! Also the view is genuinely constructivistic in the sense that the basic semantical notions, being easily definable within formalisms of sufficient complexity, are here constructed or characterized using only restricted logical, syntactical, and semantical tools. The problem of constructivism in the mathematical sense is to try to achieve by means of carefully restricted methods what is otherwise easily definable or obtainable using methods of greater power. Different kinds of constructivism arise by different restrictions upon the methods permitted. First-order constructivism, roughly speaking, allows any tools or methods or notions which are formalizable within a denumerable, first-order system.

First-order constructivism is closely allied with the philosophical view sometimes called *logical constructionalism*. This view has been especially emphasized recently by Frege, Russell, Whitehead, Carnap, Goodman, and others. Arguments on behalf of it have been put forward in Chapter I. Here we note merely that first-order constructivism may equally well be called *first-order constructionalism*.

Let us now note the main tenets of nominalism and realism, and then compare and contrast them with semantic constructivism.

First, we consider the nominalist view.[1] There are three main properties a language-system must possess to be nominalist. (1) The language-system must be a simple, applied, language-system of first order with or without identity. (2) Its rules and axioms must be such that an explicit infinity of objects of its fundamental domain is not assumed. (3) The entities of its fundamental domain should be "concrete" individuals in some sense or other, and the primitive predicate constants must apply exclusively to such individuals. Strictly speaking, these three requirements are independent of each other, in the sense that one can hold any one of them without holding the others.

In connection with (2), nominalists should not be regarded as either

[1] See especially Goodman and Quine, *op. cit.*, and Goodman, *The Structure of Appearance*, especially pp. 33–41. Their views have perhaps changed somewhat since these publications.

finitists or infinitists. We do not know whether there is a finite or an infinite number of concrete individuals in the universe. Hence to assume one or the other might be false. As a result nominalists prefer a strict agnosticism as between the infinite and the finite. With regard to (3), nominalists do not accept "abstract" entities such as classes, properties, or relations as values for variables. They do not believe in such entities, and regard as unsatisfactory any system which allows them as values for variables. The concrete individuals that are acceptable may be either physical objects or events in the spatio-temporal world, or units of experience, perhaps sensa, or qualia in some sense or other. Roughly speaking, *physicalists* regard spatio-temporal objects as the fundamental kind of entity; *phenomenalists*, units of experience or qualities in some sense. The primitive predicate constants allowed are any such as apply exclusively to the individuals considered. Of course, for specific purposes one may wish to allow only such primitive predicate constants as satisfy certain further restrictions. But any such further restrictions are not determined by purely nominalist considerations.

In terms of these three requirements various levels of nominalism may be distinguished. The strictest view, that of Goodman and Quine (in 1947), requires (1), (2), and (3) jointly. Weaker kinds of nominalism result by requiring only (1) and (2), or (1) and (3). (1) is presumably essential for any system that would be called 'nominalist' in any even remote sense. A finite set theory (without an axiom of infinity) would provide an example of a nominalism based upon (1) and (2). (The author's system of homogeneous logic[1] in one interpretation may be cited as a nominalism based upon (1) and (3).)

Of the three items, first-order constructivism requires only (1), so that first-order constructivism may be identified with nominalism in the broadest sense. But any nominalist system based upon (1), (2), and (3), or upon (1) and (2), or (1) and (3) is *eo ipso* acceptable to a first-order constructivist. Thus practically all nominalist systems are acceptable to a first-order constructivist, but many important kinds of systems, essential for formalizing various domains of mathematics and science, must be rejected by the thorough-going nominalist. Such rejection is legitimate provided such systems or theories can be reformulated in accord with his demands. But unless such methods are forthcoming, the extreme nominalist view might seem in danger of being too restrictive.

Realism, also sometimes called *platonism*, is the view that, in

[1] *The Journal of Symbolic Logic*, 8 (1943), pp. 1–23; 14 (1949), pp. 27–31; and 15 (1950), pp. 131–4.

addition to the concrete individuals of space-time or of experience, there are also abstract entities such as classes, relations, numbers, functions, etc., which can significantly be taken as values for variables. Such abstract entities or universals are, according to this view, an essential part of the "furniture of the world", and to reject them cripples science, mathematics, and language excessively. This view is that, no doubt, of most scientists and mathematicians and in one form or another is one of the most widely held positions in contemporary philosophy.

Both nominalist and realist views are usually intended to apply to all language-systems, whether as object-languages, or as syntactical or semantical meta-languages. As examples of semantical meta-languages, constructed in accord with these views, we need mention only some of the meta-languages formulated in preceding chapters. Two meta-languages have been formulated which are nominalist in the strict sense of (1), (2), and (3). ISM_D^L (Chapter XI) is nominalist in this strict sense provided the object-language L to which it is applied is also, whereas ISM_C^L (Chapter XII) is strictly nominalist whether the object-language L to which it is applied is in turn nominalist or not. ISM_C^L is thought to provide a complete solution to the problem of gaining a nominalist semantics in the strict sense of Goodman and Quine (in 1947).

As examples of realist, semantical meta-languages we can cite SM_D^S (VI,B), SM_C^S (IX,E), the various meta-languages based on satisfaction, designation, and determination as applied to realist L (VII,A–D), and indeed SM_D^L itself for realist L (IV and V). Also the various meta-languages studied by Tarski and Carnap (see (VII,E)) are of the realist kind.

The differences between first-order constructivism and realism will be discussed in the next section. As between nominalism and first-order constructivism the differences devolve upon the requirements (2) concerning finitism and (3) concerning concrete objects. With respect to (2), there would seem to be genuine gain in economy and simplicity in an L with a finite domain as over and against an L' with a denumerable one. We shall therefore adopt as a second methodological maxim that *wherever finitization is possible, it is recommended.* But the difficulty is that in some areas of science, mathematics, or philosophy, such finitization would seem impossible. E.g., a finitist system adequate for even the most elementary portions of the mathematical theory of real numbers, would seem impossible, as has in effect been suggested above. (One simple method of constructing the real numbers is to regard them as infinite classes of rationals having such and such properties. The requirement that

the classes or virtual classes of rationals be infinite seems essential.) Nonetheless a good deal more is to be expected in the direction of finitization than has heretofore been thought possible, and the further use of this second general methodological maxim might lead to some important results.

The use of 'finitism' here contrasts sharply with the use ordinarily accorded this word by mathematicians. When mathematicians speak of a finitist language they usually intend a denumerable one. A finite or finitistic language here, however, is one whose fundamental domain is taken as consisting of no more than a finite number of objects. Of course this finite number need not be specified explicitly. For strict nominalists, it is presumably the number of concrete entities in the whole of space-time. As over and against the finitist view, we shall speak for the moment of *denumerabilism* in the sense in which mathematicians ordinarily speak of finitism.

Skolem has recently put forward some comments in favor of denumerabilism as one aspect of a philosophy of mathematics.[1] Because denumerabilism would seem to stand or fall within the area of mathematical languages, Skolem's comments are especially relevant. He had hoped, he says, that the very "natural feature" of the denumerabilist view "would convince people that the . . . [denumerabilist] . . . treatment of mathematics was not only a possible one but *the* true or correct one—at least for arithmetic. Now I can well understand", he goes on, "that the lack of interest in . . . [denumerabilism] . . . is due to the circumstance that most logicians and mathematicians do not believe that it will be sufficient for mathematics. Of course this is also true, if mathematics shall mean present-day mathematics altogether, as we find it in text-books and scientific journals. It is trivial to say that present-day mathematics, which is partially built up by the aid of transfinite ideas, cannot without change be based on finitary [denumerabilist] reasoning. The question is, however, what we shall lose or gain by such a change. As to clearness and security we certainly can only gain much. . . . Now I will not be misunderstood. I am no fanatic and it is not my intention to condemn the non-finitistic ideas and methods. But I should like to emphasize that the finitistic development of mathematics as far as it may be carried out has a very great advantage with regard to clearness and security. Further [there] . . . may be good reason to conjecture that it can be carried out very far, if one would make serious attempts in that direction. I would also like to say that the

[1] Th. Skolem, "Some Remarks on the Foundation of Set Theory", *Proceedings of the International Congress of Mathematicians,* New Series (American Mathematical Society, Providence: 1952), pp. 695–704.

more importance we can attach to the purely finitistic [denumerabilist] development of mathematics, the less we need to be worried by the difficult consistency proofs of logical systems . . . containing quantifiers. The great ingenuity in the setting up of these consistency proofs, especially the proof by Gentzen and Ackermann of the consistency of formalized number theory, must be admired. But it has happened before that products of great ingenuity have lost their interest, because simpler ways of thinking have been found."[1]

Skolem's comments provide the basis of an argument in favor of the denumerabilist view as over and against the strictly finitist tendency of a thorough-going nominalism.

Let us now turn to the third tenet of nominalism concerning concrete objects. We have already remarked that the three tenets of nominalism are independent of each other. But for Goodman and Quine the second tenet (concerning finitism) is intimately bound up with their view concerning concrete objects, and moreover seems to be determined by it. Thus their view seems to be fully described by (1) and (3).

To give an exact analysis of the meaning of the third tenet of nominalism would take us far afield from semantics and syntax. To construct a language-system in which (3) could be stated would itself be a task of some difficulty. On the other hand, (1) and (2) have a more or less obvious meaning and can no doubt easily be stated within well-known formalized languages. Hence (3) lacks a kind of clarity which (1) and (2) possess. Further, (3) might be thought to be metaphysical or epistemological in nature rather than syntactical or semantical. In effect, it states something about the nature of the universe or of knowledge rather than about syntactical or semantical properties of a language. The tenet (3), taken physicalistically, would seem to issue from a prior metaphysics of space-time particles. Taken phenomenalistically, (3) seems to issue from a prior epistemology.[2] But whether syntax and semantics should rest upon any such philosophical basis may be doubted.

The attitude taken here is that syntax and semantics should be independent of any particular metaphysical or epistemological view. Semantical questions are of the kind one must consider and solve successfully prior to branching off into specific philosophical constructions. Above all, it is not the task of semantics to prefer one kind of entity to another and hence to set up prohibitions. Given entities of such and such a kind and given the expressions of a

[1] Th. Skolem, *loc. cit.*, pp. 703–4.

[2] The constructivistic system of Goodman's *The Structure of Appearance* is intended in part to provide an epistemological foundation for (3).

formalized language, the task of semantics is to study the interrelations of the two. Another argument in favor of first-order constructivism as over and against a thorough-going nominalism, then, is the rejection of (3) or of any similar philosophical provisos. But this rejection is not to be construed as anti-philosophical. Metaphysical and language-systems expressing other areas of philosophy have been treated here quite on a par with others, and the general semantical methods developed are applicable to all such language-systems alike.

It should be noted that all thorough-going nominalist systems are first-order constructivist (but not conversely) and that some realist systems are, e.g., any denumerable, first-order set theory. All nominalist systems are presumably realist, as are all first-order constructivist systems. First-order constructivism thus occupies a median position between the two other views. It has the advantages of a strict nominalism but does not reject non-nominalist, scientific language-systems of obvious importance.

D. Classes. We have already observed that, in accord with the third tenet of nominalism, concerning concrete objects, classes are flatly rejected. For thorough-going nominalists there are no such things. Realists, on the other hand, not only condone such objects but find them essential for science and mathematics. A language-system powerful enough to express the main theorems of modern mathematics without variables or other terms ranging over or designating classes or sets or other abstract objects, realists say, is inconceivable. The notion of *set* is one of the central concepts of modern mathematics. To deprive mathematics of this notion would be like depriving arithmetic of numbers. Nominalists have not succeeded in putting forward a genuine, workable alternative to this view, and hence the *onus probandi* rests with them. There can be little doubt that realists have here a cogent argument against extreme forms of nominalism. But this argument cannot be construed as an argument against first-order constructivism, in view of the possibility of denumerabilist foundations for mathematics.

None of the arguments given above (in §§ A–C) are arguments in favor of the rejection of classes. The argument concerning consistency is independent of the problem of class existence. The arguments from economy are concerned with the numbers of primitives and of objects but not with their character as abstract or concrete. Possibly arguments from simplicity can be given in favor of the rejection of classes. Possibly the notion of concrete individual in an appropriate sense can be shown to be simpler than

a notion of class (relative to a given axiom set). But this has not been shown by nominalists to date, and whether it can be done at all is far from clear. And from none of the arguments in favor of denumerable, first-order systems do we gain any methodological arguments in favor of the rejection of classes. Realist arguments concerning classes tend rather to favor first-order constructivism as against a strict nominalism as a general philosophy of language.

E. In Favor of Non-Translational Semantics. We have given several arguments in favor of first-order constructivism. The meta-language SM_C^L (Chapters VIII–IX) was formulated in detail as an example of a first-order constructivist, semantical meta-language having essentially all the deductive power required for the semantical analysis of any first-order object-language L. Similarly, ISM_C^L was formulated as an example of a nominalist semantical meta-language of commensurate power. Both these meta-languages are non-translational. Let us consider now some arguments in favor of non-translational as over and against translational semantics.

For purposes of comparison, we shall take SM_D^L as a standard formulation of translational semantics. Of non-translational systems we shall first speak exclusively of SM_C^L, and then later shall consider ISM_C^L also, which may be taken as a finitist version of SM_C^L.

In non-translational semantics there is considerable economy over translational theories in the number of primitives. In SM_C^L we need no translations of the object-language primitives at all, whereas these play an essential role in SM_D^L. Likewise there is further ontological economy in SM_C^L over SM_D^L. We have in SM_C^L, so to speak, no object-language ontology at all, having in SM_C^L no translations of the variables of L. For particular semantical purposes, use is made of the Specific Hypotheses of Comprehension of L, which together with the general theory of comprehension compensate for the drastic economy in the primitives available, as we have seen.

Another advantage of non-translational semantics is in connection with a curious circularity in the structure of translational semantics. For the discussion of this we return for a moment to the semantical meta-languages of Chapters IV–VII.

It is to be recalled again that all the translational semantical meta-languages discussed have the common feature that the object-language L reappears *in toto* within the meta-language. Every symbol of L is translated uniquely within the meta-language. Every expression of L reappears as an expression of the meta-language, every sentence as a sentence of the meta-language, every axiom or rule as an axiom or rule of the meta-language. In the proofs of consistency

of L given above in (V,E), e.g., we notice that the semantical tools employed include all the deductive tools of the object-language, and in addition much more. We presuppose in the semantical meta-language all the means of expression, all the axioms, and all the rules of inference of the very language with whose consistency we are concerned. Unless one has some prior reason for thinking the object-language consistent, one might not be especially convinced by such a consistency proof. For if L contained some hidden contradiction of which we were not aware, the semantical meta-language would contain it also. Consistency proofs in translational semantics seem circular in the sense that we allow ourselves the use of translations of the deductive procedures of the very language whose consistency we seek to prove. Of course this does not constitute a vicious circularity and interest attaches to such consistency proofs in spite of it. But this circularity makes us uncomfortable nonetheless. It leads us to think that perhaps semantical consistency proofs purport to accomplish more than they actually do. The details of such proofs might strike us as mathematically interesting, but they do not seriously convince us of consistency. In short, such proofs might seem to be interesting logic or mathematics, but they are not interesting *meta*-logic or *meta*-mathematics. In these latter disciplines we wish to be convinced, using only very sparse deductive procedures.

This curious circularity seems to be a disadvantage of employing semantical meta-languages for the purposes of proving consistency. To provide a consistency proof of L is one of the important purposes for constructing a semantical meta-language of L at all, as we have already noted. And if now we find that the purported consistency proof is unconvincing, some of the importance of systematic semantics might seem to be lost.

We must now inquire into the sense in which, if any, non-translational semantics can be said to avoid this curious circularity. First, we recall that no reference is made to the objects of L within SM_C^L by means of variables ranging over them. Within SM_C^L one speaks exclusively of the *expressions* of L and interrelates them in certain ways. Further, in carrying out the consistency proof of L within SM_C^L, use is made of the Specific Hypotheses of Comprehension of L as hypotheses wherever needed. In this way we do not commit ourselves as to whether the Specific Hypotheses hold or not. If they hold, then we show that L is consistent. We know from the relative consistency proof of the General Rules of Comprehension that all the tools permitted in the proof of the semantical consistency of L (other than the Specific Hypotheses of Comprehension) are secure relative to syntax. Hence any doubt that can arise concerning the

consistency proof of L must be referred to syntax or to the Specific Hypotheses of Comprehension. In this twofold way, that the entities of L are not values for variables and that the Specific Hypotheses of Comprehension are taken as additional hypotheses wherever needed, this curious circularity of translational semantics is avoided within SM_C^L.

But of course it must not be supposed that this circularity is avoided in the sense of dispensing with the need for the Specific Hypotheses of Comprehension. Nor indeed would it be desirable to dispense with such hypotheses, because they are determined by the axiomatic character of the object-language. The semantics of L is primarily the semantics of the axioms of L, and it is by means of the Specific Hypotheses that we gain the semantical description or characterization of the axioms. And use of the Specific Hypotheses is essential for the proof of consistency of L. A kind of residuum of the curious circularity of consistency proofs in translational semantics thus remains in non-translational semantics. Some such circularity seems essential to all known semantical methods of proving consistency.

An especially important advantage of SM_C^L over SM_D^L is to be noted in connection with the first methodological maxim. SM_C^L is a denumerable first-order language whether the object-language L to which it is applied is or not, whereas the denumerability of SM_D^L rests upon that of L. Hence all the advantages of denumerable first-order systems accrue to SM_C^L as against SM_D^L for first-order L in the wider sense.

SM_D^L was shown consistent relative to the axioms of M together jointly with the axioms of L. But SM_C^L was shown consistent relative to the axioms of M only. The consistency of SM_C^L is therefore in this sense more secure than that of SM_D^L. This constitutes another advantage of SM_C^L over SM_D^L. Still another is that SM_C^L makes no use of the Rule of Infinite Induction, *SynR4*. Proofs in SM_C^L are of the ordinary, finite kind, whereas proofs in SM_D^L may contain an infinity of premisses.

The inscriptional meta-languages ISM_D^L and ISM_C^L may be taken as finitistic versions respectively of SM_D^L and SM_C^L. ISM_C^L has the same advantages over ISM_D^L that SM_C^L has over SM_D^L. These have been discussed above. Let us now consider what advantages if any ISM_C^L has over SM_C^L, or for that matter ISM_D^L over SM_D^L.

ISM_C^L is a clear-cut nominalism. It is of first order, may be finitistic, and the values for variables are exclusively concrete objects. It is constructed in accord with the second (as well as the first) methodological maxim. The difficulties noted above in the way of the

general thesis of finitism do not apply here. In ISM_C^L we have a satisfactory semantical meta-language of an extremely restricted kind. There is no need, for present purposes at least, for a semantical meta-language of greater complexity or power of expressiveness. ISM_C^L is probably the most restricted and simplest semantical meta-language that has yet been formulated. And its character as such in no way depends upon L, which may be taken as any first-order language of whatever complexity.

ISM_D^L, on the other hand, depends fundamentally upon L, and is a nominalist system if and only if L is. ISM_D^L may be finitistic if and only if L is. But SM_D^L is not interpretable finitistically under any circumstances. ISM_D^L has the advantage over SM_D^L of being formulated in accord with the second as well as with the first methodological maxim, for finite L.

F. Minimality. We now show that non-translational semantics provides a *minimal* semantics in an appropriate sense.

First we must clarify what is to be understood by a minimal semantics, and justify the definition proposed on intuitive grounds. Speaking very roughly, a minimal semantics is one presupposing as little as possible in the choice of primitives and of objects taken as values for variables. Also, as little as possible should be assumed in the axioms and rules. Hence for a semantics to be minimal we should no doubt wish it to be non-translational rather than translational. A semantics must contain a truth-concept and hence of course all concepts definable in terms of it. But these concepts in effect exhaust semantics, in the extensional sense in which the word is used in this book. We are perhaps justified therefore in defining a minimal semantical meta-language as one having these two properties, i.e., as one which (i) is non-translational, and which (ii) contains a non-translational truth-predicate together with all and only those semantical notions appropriately definable in its terms.

That this definition is intuitively adequate may be seen as follows. We have noted in the preceding sections the advantages of non-translational semantics with regard to the number and kinds of primitives and with regard to the number and kinds of objects presupposed as values for variables. (Only very little if any further reduction is possible in these directions if the semantics is finitistic.) We have as yet no precisely defined notion as to what is meant when we speak of "assuming as little as possible" by means of axioms or rules. There is a need for a precise characterization of this notion. But however this be defined, it is perhaps less important to assume little by way of axioms and rules than it is to be sure that what we

assume is consistent or relatively consistent. If, now, as in the case of the semantical axioms of non-translational semantics, we can show the axioms consistent relative to the preceding theory by interpreting, in a very simple way, the primitive 'Cmprh' within that theory so that all the rules concerning it become provable, we might in a sense be said to be assuming nothing new at all. Because, in this case, the new axioms already have an interpretation or model within the preceding theory. Hence it is difficult to see how less could be assumed by way of specifically semantical axioms. It seems reasonable therefore to require (i) in the definition of a minimal semantics. But it is also reasonable to require (ii), 'Tr' and the notions definable in terms of it exhausting semantics in the sense used here.

We have seen in (VIII,E) that 'Cmprh' and hence all other notions of non-translational semantics are definable in terms of 'Tr'. Therefore nothing is definable in SM_C^L or any of its alternatives not definable within a non-translational meta-language for L with 'Tr' as primitive. Hence no semantical predicates are definable within a non-translational semantics not definable in terms ultimately of 'Tr'. Hence each non-translational meta-language satisfies the requirement (ii) in the definition of a minimal semantics and the definition may be simplified by dropping (ii) altogether. This observation clearly amounts to a "proof" that any non-translational semantics is minimal.

In these considerations concerning minimality we have not distinguished between classical and inscriptional theories. Enough has already been said concerning the interrelations between these. Little would be gained by requiring a minimal semantics to be also inscriptional or finite, although this can be done if desired. The importance of finitism and of inscriptional theory has been indicated sufficiently already. Also, finitude is a property of the underlying syntax rather than of the semantical superstructure. We do not require of a minimal semantics that it be based upon a minimal syntax (in some sense). We require only that it be based upon a first-order syntax of the kinds discussed in Chapters III or XI.

Also we do not require that a minimal semantics be the simplest. A semantics based on a one-place predicate (such as 'Tr' or 'Univ') is no doubt simpler in an appropriate sense than one based upon a non-symmetrical two-place predicate (such as 'Cmprh'). We have seen in (VIII,E) that the various formulations of non-translational semantics are in fact equivalent. We therefore regard them all as minimal. But considerations concerning the comparative simplicity of the primitive predicates of two systems and concerning the

comparative simplicity of their axioms and rules might well lead us to prefer one to the other.

G. On Understanding a Language-System. According to Carnap, we can say we "*understand* a language-system, or a sign, or an expression, or a sentence in a language-system, if we know the semantical rules of the system. We shall also say that the semantical rules give an *interpretation* of the language-system." [1] And again, ". . . to understand a sentence, to know what is asserted by it, is the same as to know under what conditions it would be true ".[2] The semantical rules of a language are such as to "determine a truth-condition for every sentence of the object-language, i.e., a necessary and sufficient condition for its truth ".[3] These truth-conditions are presumably extra-linguistic, i.e., refer to non-linguistic facts in some sense or other. Let us consider now what it means to say we understand, in this sense, a language-system whose semantics is formulated within SM_D^L or SM_C^L. *Mutatis mutandis*, these remarks will apply equally well to the other methods developed.

The semantical rules *DenR1* and *DenR2* of SM_D^L do provide a necessary and sufficient condition under which a given sentence of L is true. E.g., '$(x)(\mathrm{P}x \text{ .v. } \sim\mathrm{P}x)$' is true if and only if

(1) '$x\ni(\mathrm{P}x \text{ .v. } \sim\mathrm{P}x)$' denotes every object,

and this holds in turn by an Adequacy Rule, if and only if

(2) For every object x, either x has P or x does not have P.

(1) here is a rewriting of the statement that '$(x)(\mathrm{P}x \text{ .v. } \sim\mathrm{P}x)$' is true, and does not describe an extra-linguistic fact. But (2) does provide an extra-linguistic, necessary, and sufficient condition under which '$(x)(\mathrm{P}x \text{ .v. } \sim\mathrm{P}x)$' can be said to be true. The presence of the translations of the object-language variables in SM_D^L, where they are regarded as ranging over the same objects as the variables of L, assures the necessary connection with objects. That the semantical truth-predicate defined in SM_D^L is adequate gives the necessary and sufficient condition under which a sentence of L is true. But the adequacy condition is provable on the basis of *DenR1* and *DenR2*. Hence to know *DenR1* and *DenR2*, together of course with the logical and syntactical rules of SM_D^L, is to understand L.

When we turn to the non-translational meta-language SM_C^L we find a quite different situation. Here we have no translations of the primitives of L. The statements of SM_C^L speak only of the expres-

[1] R. Carnap, *Foundations of Logic and Mathematics*, pp. 10–11.
[2] R. Carnap, *Introduction to Semantics*, p. 22.
[3] R. Carnap, *loc. cit.*

sions of L and not of non-linguistic objects. The semantical rules of SM_C^L refer only to expressions of L and hence seem to afford no connection with extra-linguistic fact. What does it mean then to say that we understand a language-system L if its semantics is formulated within SM_C^L?

To answer this we must consider more closely the different ways of construing the semantical primitive 'Cmprh' of SM_C^L. Within SM_D^L we defined (in (IV,C))

'a Cmprh b' as '(PredCon a . PredCon b . $(x)(b \text{ Den } x . \supset . a \text{ Den } x)$)'.

Within SM_D^L if 'a Cmprh b' holds then a denotes every object *of the fundamental domain of* L which b does. Turning now to SM_C^L, we can construe 'Cmprh' in precisely the same way. Given L we can construe 'a Cmprh b' to express that a denotes every object of the fundamental domain of L which b does. Of course we cannot make this statement within SM_C^L. But the important point is that we can regard 'a Cmprh b' within SM_C^L as expressing *precisely* what it does within SM_D^L. To say that a Cmprh b in SM_C^L means that a comprehends b in such and such a non-null domain of individuals. For each object-language L one expects ordinarily to have a multiplicity of different possible interpretations. To provide these different interpretations different relations of comprehension are required, each providing an interpretation within an appropriate domain.

Construing 'Cmprh' in this way, we see that within SM_C^L we can refer implicitly to the objects of L, even though we cannot do so explicitly by means of variables ranging over them. But this implicit reference to the objects of L is all that is required. By means of it we gain a full semantical description of L, and in particular know precisely what objects the constituent expressions talk about.

We can say we understand L in Carnap's sense by means of the method of SM_C^L by knowing the semantical rules of SM_C^L. We can say also that the semantical rules provide an interpretation for L. The implicit connection with extra-linguistic fact provided by these rules does not give a non-linguistic, necessary and sufficient condition under which a sentence can be said to be true, because the adequacy condition does not explicitly obtain. Within SM_C^L we can only form other statements about L as necessary and sufficient conditions for the statement that a given sentence of L is true. But this limitation does not mean that we cannot be said to understand the given sentence of L. Construing 'Cmprh' in the appropriate way we see that SM_C^L provides a genuine semantics in essentially the same sense as that provided by ordinary, translational semantics.

To see this a little more clearly, let us glance again at $SM_C^{L'}$

(Chapter X), where the object-language contains one or more primitive individual constants. From one of the theorems of this metalanguage, ($TE6b$(X)), one can prove immediately that

$$\vdash \mathrm{Tr}\ (pee \frown ay)\ .\equiv.\ ay\ \mathrm{Sub}^{oo}\ pee,$$

in other words, that 'Pā' is true in L if and only if ay bears Sub^{oo} to pee. From another semantical theorem ($TE5$(X)) we know that ay is a Unit^{oo} and this means in effect that ay denotes one and only one object of the fundamental domain of L. To know that Tr ($pee \frown ay$), then, we need merely note that the object which ay denotes is denoted also by the predicate pee. This is in effect to verify that the object which ay denotes has a certain property. To know that 'Pā' is true is in effect therefore to know that the individual ā has the property P.

We might construe 'Cmprh' somewhat more broadly, so that 'a Cmprh b' expresses that a denotes whatever objects b does without specifying the kind of objects. A semantics based upon 'Cmprh' construed in this sense would give only a *partial* interpretation of L. We should never know from the semantical rules precisely what entities are talked about in L. But on the other hand the semantical rules would tell us a good deal concerning the way in which the various words (abstracts) of L are interrelated, i.e., that whatever is denoted by one is denoted by such and such another, etc. A partial interpretation of this kind would suffice for many semantical purposes, e.g., for a consistency proof, but it is doubtful whether such a semantics would contain a genuine truth-concept in the intuitive sense. But we have already seen that SM^L_C provides a full interpretation for L, construing 'Cmprh' in the narrower sense, so the notion of a partial interpretation need not concern us further.

Among the various relations of comprehension some are less extensive than others, owing to differences in the numbers of objects implicitly referred to. E.g., a relation of comprehension in the domain of natural numbers is less extensive than such a relation in the domain of real or complex numbers. Also a relation of comprehension whose field consists of a finite number of objects of a given kind is less extensive than one whose field consists of a larger finite number of or an infinite number of objects of the same kind. E.g., the relation of comprehension whose field consists of natural numbers ≤ 10 is less extensive than the comprehension relation in the domain of all natural numbers. To return to a point made at the end of § A above, the non-translational semantics based on comprehension of an object-language talking about a finite number of objects of a certain kind can perhaps be regarded as simpler (in some appropriate sense) than

the non-translational semantics based on comprehension of an object-language talking about an infinite number of objects of the same or of a comparable kind, because the relation of comprehension of the former is less extensive than that of the latter.

Finally, we must consider the question as to whether a non-translational semantics should properly be called a semantics at all. If we think of semantics as being concerned with the relations between the linguistic expressions of a language and the (usually non-linguistic) objects about which the language speaks, then non-translational semantics contains no relations which are properly semantical at all. The relations which it does contain are definable in terms of properly semantical relations but have only expressions as relata. Such relations might be regarded as purely syntactical. Also, if we interpret 'Cmprh' in the broader way suggested above, non-translational semantics provides only a partial interpretation of the object-language and hence does not give a genuine semantics at all.

We have already noted that by construing 'Cmprh' in the narrower sense we do gain a full interpretation for the object-language within SM_C^L in essentially the same way as in SM_D^L. Also such an interpretation gives a genuine semantical truth-concept. Also various important relations of SM_C^L are definable in terms of the properly semantical relation of multiple denotation (IV,C). Being definable in terms of properly semantical relations, they may be regarded as semantical even though their relata are exclusively expressions. This may be seen more clearly if we consider for a moment the non-translational semantical meta-language based on 'Tr' in (VIII,E). Interpreting 'Tr' in the usual sense as a truth-concept, the resulting meta-language is clearly semantical. Although 'Tr' applies only to expressions, we should not therefore regard it as a syntactical predicate. The situation here is similar to that of 'Cmprh'.[1] Hence we are justified throughout in speaking of non-translational *semantics* and of SM_C^L, or $SM_C^{L\prime}$, and of ISM_C^L as *semantical* meta-languages.

We shall return to the discussion of first-order constructivism in § I below, which presupposes the material to be developed in the next section.

H. Toward a Formal Theory of Semantical Systems. The semantical meta-languages developed in this book have been *general*, as we have already noted, in the sense that given L of the appropriate kind the semantics of L can be formalized within such meta-languages. None of the semantical meta-languages, however, is general

[1] The author is indebted to Professor W. V. Quine for calling attention to this similarity.

in the sense of being applicable to any L, *for variable 'L'*. The various semantical theories developed above are applicable only where L is in a certain sense specified in advance. A general semantics, in the sense of a semantics applicable arbitrarily to any formalized language-systems (of such and such kinds) as values for variables, remains yet to be formulated.[1] We shall sketch somewhat roughly and tentatively such a theory in this section, as well as a slightly weaker kind of theory in which languages are identified with virtual classes or relations.

To make the kind of generality we are seeking precise we need to introduce variables over, and virtual class expressions for, formalized language-systems both with and without specified interpretation. For this the notion of being a formalized language-system needs further analysis and clarification. This notion was vaguely indicated in (I,A), and many examples of formalized languages have been given throughout. But we must now be much more precise.

Before considering more closely the notion of an interpreted language or semantical system, let us consider that of a logistic system or uninterpreted calculus. In (I,A) we noted that such a calculus comprises six items, the primitive vocabulary, a definition of 'formula' and possibly 'term', axioms, rules of inference, theorems, and definitions. Definitions we have regarded throughout as conventions of abbreviation which simplify the notation and therefore are not essential. Also, under suitable circumstances, the notion of term can be dispensed with, as we know from (II,F). Suppose, then, that we have just the five remaining items. We might tentatively regard these as definatory of an uninterpreted language-system. But we can reduce these even further. For this we must think of syntax in a somewhat broader sense than was done in Chapters III or XI or XII.

For a general theory of logistic systems as values for variables (or as extensions of expressions for virtual classes), we must presuppose a syntactical framework in which we can talk about more than one object-language. In Chapters III and XI and XII all the syntactical notions introduced were relativized to a single object-language. We must thus first consider the changes required, say, in Chapter III. (Those required for Chapters XI or XII are not drastically different.) Very briefly we may outline these as follows.

Consider a finite or infinite number of object-languages, L_1, L_2,

[1] Cf., however, A. Tarski, "Grundzüge des Systemenkalküls", *Fundamenta Mathematicae*, 25 (1935), pp. 503–26, and 26 (1936), pp. 283–301; and K. Schröter, "Was ist eine Mathematische Theorie?" *Jahresbericht der Deutschen Mathematiker-Vereinigung*, 53 (1943), pp. 69–82.

and so on, with or without identity. For simplicity we can assume that they contain no primitive functional or individual constants, as we have frequently done above. As primitives in the meta-language we need just the structural-descriptive names of Chapter III. Each primitive predicate constant of the object-languages can be expressed by means of 'P' followed by a specified number of accents. The narrow vocabulary of Chapter III will thus suffice to express simultaneously the syntactical properties of any finite or infinitely large class of first-order object-languages. For the moment we do not distinguish among these languages but consider only the syntactical properties of the various symbols. To fix these properties we assume by way of axioms appropriate adaptations of *SynR1–SynR3*, *TG4*(III), and *TG13*(III). The resulting theory is elementary syntax generalized. Let this language be called *GM*, the generalized (syntactical) meta-language. The values for variables are the expressions and concatenates of expressions of all the *L*'s under consideration, all mixed up in such a way that one does not distinguish the expressions of one language from those of another.

The syntactical definitions in Chapter III we wish to retain in *GM*. In particular, definitions of 'Bgn', 'Ends', 'FrIng', etc., are to be retained precisely as given. But of course the variables now range over all the expressions of the *L*'s indiscriminately, and not over just the expressions of some fixed *L*. Some of these notions, in particular 'PrimPredCon', 'AtFmla', 'Fmla', 'PredCon', and notions defined in terms of them, we shall wish to retain in this general sense but *also* we shall wish to relativize them to a given *L*. We can let 'PrimPredCon$_n$ *a*' express within *GM* that *a* is a primitive predicate constant of *L* . 'PrimPredCon *a*' expresses, if we consider an infinite number of languages, that *a* is a *pee* or a *pee* followed by an AcString. But 'PrimPredCon$_n$ *a*' is defined by enumerating just the primitive predicate constants of L_n. Similarly let 'Ax$_n$ *a*' express that *a* is an axiom of L_n. And so on for the other relativized notions. (Of course the formulae of L_n may in fact be the same as those of L_m. This is permitted. Also the axioms of L_n may be the same as those of some L_m. We shall consider in a moment the circumstances under which two *L*'s are equivalent.) In this way we can formulate the syntax of each *L* within *GM*. Note that the relation IC of immediate consequence is not relativized, its definition in no way depending upon the relativized notions of formula, predicate constant, or the like.

We saw a moment ago that in a tentative way we could regard a logistic system or uninterpreted language as being determined by five items, the primitive vocabulary, the formulae, the axioms, the theorems, and the rules, and it was remarked that these could be

further reduced. In fact, given just 'Ax_n', the other notions in the syntax of L_n are uniquely determined. We shall then be able to identify L_n with *the virtual class of its axioms.*

Let us note how, given 'Ax_n', the other items definatory of L_n are uniquely determined. First consider 'AtFmla_n'. Every primitive predicate constant of L_n will occur in some axiom of L_n, and furthermore occur only with the appropriate number of variables as arguments. That a is a *string of variables* may be defined as follows.

'VblString a' may abbreviate '$(\mathrm{E}b)(a\ \mathrm{FrIng}\ b\ .\ (c)(c\ \mathrm{FrIng}\ b :\supset : \mathrm{Vbl}\ c\ .\mathrm{v}.\ (\mathrm{E}d)(\mathrm{E}e)(d\ \mathrm{Pr}_b\ c\ .\ e\ \mathrm{Pr}_b\ c\ .\ c = (d \frown e))))$'.

Then 'AtFmla_n' is definable in terms of 'Ax_n'.

'$\mathrm{AtFmla}_n\ a$' abbreviates '$(\mathrm{E}b)(\mathrm{E}c)(\mathrm{E}d)(\mathrm{PrimPredCon}\ b\ .\ \mathrm{VblString}\ c\ .\ a = (b \frown c)\ .\ \mathrm{Ax}_n\ d\ .\ a\ \mathrm{Occ}\ d\ .\ \sim (\mathrm{E}a')(\mathrm{Vbl}\ a'\ .\ (b \frown c \frown a')\ \mathrm{Occ}\ d))$'.

(We presuppose that '$\mathrm{Ax}_n\ d$' is already defined in *GM* wholly in primitive terms, so that no circularity is involved here. Or, otherwise, we may transform this definition into a material equivalence, which is then presumably provable.) If '=' is a logical primitive of L_n a clause must be added in the definiens here concerning atomic identity formulae. 'Fmla_n' may then be defined by framed ingredients as in (III,D). The uniqueness of this notion may easily be shown.

The relation of being an immediate consequence of in L_n is merely IC as confined to the formulae of L_n.

'$a\ \mathrm{IC}_n\ b,c$' abbreviates '$(a\ \mathrm{IC}\ b,c\ .\ \mathrm{Fmla}_n\ b\ .\ \mathrm{Fmla}_n\ c)$'.

Also the theorems of L_n are just the formulae which can be reached in a finite number of IC_n steps, so to speak, from the axioms of L_n. This notion is definable in *GM* by framed ingredients essentially as in (III,F).

Let 'PS a' express within *GM* that a is a primitive symbol of any one of the languages L_1, L_2, etc. '$\mathrm{PS}_n\ a$' can then be defined simply as 'PS a', for each n, the primitive symbols of all the L's being the same. (Recall here that 'P′', e.g., is not a primitive symbol, but a concatenate.)

What then, precisely, is an uninterpreted language-system? Within a general syntactical framework such as *GM*, each L_n may be identified with the notion expressed by 'Ax_n', and all the items definatory of L_n are thereby uniquely determined. The first of the two general syntactical methods mentioned is to identify L with the *virtual* class of its axioms. (Note that a virtual class of axioms is not a value for a class variable in *GM*, this meta-language containing no such variables.) We see from the foregoing that the syntax of each first-order language may be handled on the basis of such an identi-

fication. Also within *GM* we can interrelate the syntax of the various object-languages, thereby achieving a basis for comparative syntax.

The second method of achieving a general syntax is to introduce appropriate class variables within *GM*. Rather than to presuppose a basic logic of first order for this, we must employ one of second order. Logistic systems are then identified with the proper kinds of classes, and hence are values for variables. Here also we should have a basis for the comparative syntax of the various object-languages considered. Let this general syntax-language be called *GM'*. *GM'* differs from *GM* primarily in presupposing a more powerful underlying logic.

The main advantage of the method of *GM'* over that of *GM* is that within *GM'* we can *quantify* over logistic systems, and hence formulate theorems of the form 'There exists a logistic system having such and such a property' or 'All logistic systems have such and such a property'. On the other hand, *GM* is much simpler, is of first order with a denumerable fundamental domain of objects, etc., and hence has many of the advantages enumerated at the beginning of this Chapter. In *GM* we cannot quantify over logistic systems, but instead must introduce a virtual class expression for each one as needed.

(Both *GM* and *GM'* may be transformed into inscriptional meta-languages by taking inscriptions rather than shapes as values for the expressional variables. In the case of *GM* this would presumably yield a genuine, perhaps finitistic nominalism. But the theory which would result in this way from the other meta-language *GM'* might not appear very interesting. It would clearly not be nominalist, and shapes would reappear as values for class variables. Nonetheless such a meta-language might prove to be of interest for other reasons.)

Let us now turn again to semantics and semantical systems, and let us think of semantics in the sense of non-translational semantics. We can take as a basis for the discussion any one of the equivalent formulations of non-translational semantics. Let us take SM^L_C, which has been developed in some detail.

When we have occasion to speak of more than one relation of comprehension it is convenient to introduce a subscript indicating the domain of individuals implicitly referred to. Also, to indicate the relativization to a particular language-system we shall use numerical superscripts.

$$\text{'}a \text{ Cmprh}^n_{D_m} b\text{'}$$

is thus to express that a comprehends b in the non-null domain of individuals D_m and in the language-system L_n. Within a general semantics we shall need several relations of comprehension in various

domains and in various language-systems. In order to construct a general semantical meta-language upon the basis of *GM*, for a finite number of the systems L_1, L_2, etc., expressions for these several relations of comprehension may be taken as primitive. Let *GSM*, the generalized semantical meta-language, result from *GM* by the addition of a finite number of primitives for the various relations of comprehension needed. Concerning each such relation axioms akin to *GRC1–GRC14* are assumed, and in terms of each such relation definitions of the various fundamental semantical concepts in (VIII,B and C) may be given, including a semantical truth-concept. These notions are relativized to a given L_n, the definitions being given in terms of the appropriate '$\mathrm{Cmprh}^n_{D_m}$'. Specific Hypotheses of Comprehension concerning each $\mathrm{Cmprh}^n_{D_m}$ may be defined, stating in effect that the non-logical axioms of L_n, if sentences, or closures of such, are true (in the sense appropriately determined by $\mathrm{Cmprh}^n_{D_m}$). In this way we gain within *GSM* a full semantical description of the language-systems L_n.

Among the various primitive relations of comprehension of *GSM*, we may in fact have more than one relation referring to the same L_n but to different domains. E.g., '$\mathrm{Cmprh}^n_{D_1}$' and '$\mathrm{Cmprh}^n_{D_2}$' might both be primitives of *GSM* for D_1 and D_2 distinct. This is allowed by the foregoing explanations. Using such relations of comprehension we can construct different interpretations for the same logistic system. In this way we can compare and contrast within *GSM* different interpretations of the same system as well as consider interpretations in the same domain of individuals of differing logistic systems. Within *GSM* we can therefore lay the foundation for comparative semantics. For this further semantical axioms would no doubt be required, connecting the various relations of comprehension in appropriate ways.

As an example of a theorem in comparative semantics of especial importance, let us glance for a moment at a famous meta-mathematical theorem due to Löwenheim.[1] Suppose we are given a non-null domain D of objects and a sentence a of one of the first-order language-systems L_n. We can think of a as written wholly in primitive terms and hence as containing one or more occurrences of the primitive predicate constants of L_n. (To simplify, let us suppose that the identity sign is omitted as a logical primitive, and that if admitted at all it is one of the non-logical primitive predicate constants.) We can then say that the sentence a is *satisfiable* (erfüllbar) or *true in the*

[1] L. Löwenheim, "Über Möglichkeiten im Relativkalkül", *Mathematische Annalen*, 76 (1915), pp. 447–70.

domain D_m if and only if a is true in the sense of 'true' defined for the sentences of L_n in terms of '$\mathrm{Cmprh}^n_{D_m}$'.[1] In other words, a can be said to be true or satisfiable in D_m where we regard the variables of a as ranging over the objects of D_m and where the primitive predicate constants of a are given specific designata as properties of or relations between or among the objects of D_m.

The so-called Löwenheim theorem states in effect that any sentence of a language-system of first order is satisfiable in a domain at most denumerable, if it is satisfiable in any non-null domain whatsoever, finite, denumerable, or non-denumerable. Löwenheim's proof requires fundamentally a very complicated use of subindices as well as of expressions presumed to contain a non-denumerably infinite number of terms. (Several simpler and more perspicuous proofs of this theorem have been given by Skolem.[2] The first such proof (1920) made fundamental use of the axiom of choice.[3] The use of the axiom of choice together with a certain recursive process in effect supplants Löwenheim's complicated use of subindices. In 1922 and 1929 Skolem gave other proofs along mathematically constructivist lines and in which the use of the axiom of choice was avoided.[4])

Let '$\mathrm{Tr}^n_{D_m}$' symbolize the truth-concept for L_n defined within *GSM* in terms of '$\mathrm{Cmprh}^n_{D_m}$'. Within *GSM* the Löwenheim theorem states that every formula of the form

$$\text{'}\mathrm{Tr}^n_{D_m}\, a \,.\supset.\, \mathrm{Tr}^n_{D_0}\, a\text{'}$$

is provable, where D_0 is the domain of natural numbers. Presupposing sufficiently powerful semantical axioms, one can perhaps mirror one or more of Skolem's proofs within *GSM*.

Let us now consider the kinds of objects within *GSM* with which semantical systems are to be identified. Logistic systems we identified in *GM* with virtual classes. Semantical systems involve also a truth-concept or a relation of comprehension. A relation of

[1] Cf. Hilbert and Bernays, *Grundlagen der Mathematik*, vol. 1, p. 128 and *passim*. Also A. Tarski, *Der Wahrheitsbegriff*, pp. 318–27.

[2] See Th. Skolem, "Logisch-kombinatorische Untersuchungen über die Erfüllbarkeit oder Beweisbarkeit Mathematischer Sätze nebst einem Theoreme über Dichte Mengen", *Skrifter utgit av Videnskapsselskapet i Kristiania*, 4 (1920); "Einige Bermerkungen zur Axiomatischen Begründung der Mengenlehre", *Wissenschaftliche Vorträge gehalten auf dem Fünften Kongress der Skandinavischen Mathematiker in Helsingfors vom 4. bis 7. Juli 1922* (Helsingfors: 1923), pp. 217–32, and "Über einige Grundlagenfragen der Mathematik", *Skrifter utgitt av Det Norske Videnskaps-Akademi i Oslo* (I. Math.-naturw. klasse 1929, no. 4). These papers will subsequently be referred to as Skolem (1), Skolem (2), and Skolem (3) respectively.

[3] Skolem (1), pp. 6–11.

[4] Skolem (2), pp. 220–1, and Skolem (3), pp. 23–8.

comprehension within *GSM* is also strictly a virtual relation. We can, then, identify a semantical system with a *virtual ordered couple of a virtual class* with a *virtual relation.* Expressions for virtual ordered couples may be introduced within *GSM* by iterating the process whereby we introduced expressions for virtual relations in (II,D). We can define, it will be recalled,

$$\text{`}a'b'c' \varepsilon abc \ni X\text{' as } X',$$

where X' differs from X in the way required, thereby gaining expressions

$$\text{`}abc \ni X\text{'}$$

for virtual three-place relations. Let now 'F_1^k', 'F_2^k', etc., stand for any expressions for k-place virtual relations. We can then define

$$\text{`}F_m^n F_k^q \ni X\text{'}$$

in context as follows.

'$F_o^n F_p^q \varepsilon F_m^n F_k^q \ni X$' is defined as X', where (i) F_m^n and F_k^q are any distinct expressions for virtual relations containing no free variables and not occurring in the formula X', (ii) F_o^n and F_p^q contain no free variables, and (iii) X differs from X' only in containing F_m^n and F_k^q in 0 or more places where there are occurrences of F_o^n and F_p^q respectively in X'.

Expressions for semantical systems may be identified with expressions of the form

$$\text{`}F_1^1 F_1^2 \ni X\text{'}$$

or of the form

$$\text{`}F_1^1 F_2^1 \ni X\text{'}.$$

More particularly, an interpreted L_n may be identified with a virtual ordered couple of a virtual class $a \ni \mathrm{Ax}_n\, a$ and a virtual two-place relation $ab \ni a\ \mathrm{Cmprh}_D^n\, b$, or with a virtual ordered couple of $a \ni \mathrm{Ax}_n\, a$ with the virtual class $a \ni \mathrm{Tr}_D^n\, a$. '$\mathrm{Tr}_D^n$' and '$\mathrm{Cmprh}_D^n$' are interdefinable. The treatment of semantical systems will be a little simpler if we make the second identification here according to which a semantical system L_n with fundamental domain D is regarded as the ordered couple for which

(1) $$\text{`}F_1^1 F_2^1 \ni (a)((F_1^1\ a\ .\equiv.\ \mathrm{Ax}_n\, a)\ .\ (F_2^1\ a\ .\equiv.\ \mathrm{Tr}_D^n\, a))\text{'}$$

stands, where F_1^1 and F_2^1 are any one-place abstracts containing no free variables chosen in advance. This expression (1) may conveniently be abbreviated as

(2) $$\text{`}\langle \mathrm{Ax}_n, \mathrm{Tr}_D^n \rangle\text{'},$$

and

$$\text{`}SL_n\text{'},$$

read 'the semantical system L_n', may then be introduced as an abbreviation for (2).

A second method of achieving expressions for semantical systems is to introduce appropriate class variables and quantifiers over them of two higher types. The new variables and quantifiers are introduced to handle classes of expressions and to handle ordered couples of such classes. Let the meta-language formulated in this way be called *GSM'*. Within *GSM'* a semantical system is to be identified with an ordered couple of a class of axioms and a class of truths. The advantages and disadvantages of *GSM'* relative to *GSM* are similar to those of *GM'* relative to *GM*.

In closing this rough sketch of the two general syntactical and semantical methods, let us consider various circumstances under which two logistic systems might be said to be *equivalent*.

Within *GM* we can let

$$\text{'}L_n\text{' abbreviate '}\mathrm{Ax}_n\text{'}.$$

We can now define within *GM* several types of *syntactical* equivalence. First

$$\text{'}L_n \text{ AxEquiv } L_m\text{' may abbreviate '}(a)(L_n\, a \,.\equiv.\, L_m\, a)\text{'}.$$

Two logistic systems are said to be *axiomatically equivalent* if and only if their axioms coincide. But also we may let

$$\text{'}L_n \text{ ThmEquiv } L_m\text{' abbreviate '}(a)(\mathrm{Thm}_n\, a \,.\equiv.\, \mathrm{Thm}_m\, a)\text{'}.$$

Two systems are said to be *equivalent with respect to theoremhood* if and only if every theorem of one is a theorem of the other and conversely. Also two systems may be said to be *equivalent with respect to their formulae* if and only if the formulae of one are the same as the formulae of the other.

$$\text{'}L_n \text{ FmlaEquiv } L_m\text{' abbreviates '}(a)(\mathrm{Fmla}_n\, a \,.\equiv.\, \mathrm{Fmla}_m\, a)\text{'}.$$

Obviously, in view of these definitions, we have that

TH1. $\vdash L_n \text{ AxEquiv } L_m :\supset: L_n \text{ ThmEquiv } L_m \,.\, L_n \text{ FmlaEquiv } L_m.$

But we might have two logistic systems equivalent with respect to their formulae, but not with respect to their axioms or theorems. Also

TH2. $\vdash L_n \text{ ThmEquiv } L_m \,.\supset.\, L_n \text{ FmlaEquiv } L_m.$

But we might well have two logistic systems equivalent with respect to their theorems but not axiomatically equivalent.

There are other types of syntactical equivalence one could consider but these appear to be the most important. Also there are several types of syntactical *isomorphism* between systems which may be introduced. This may be done in *GM* only with certain

restrictions. For definitions of isomorphism we should presumably wish to employ the more powerful logical devices of *GM'*.

Finally, let us consider within *GSM* a few relations of equivalence between *semantical* systems. Ordinarily an ordered couple $\langle M,N\rangle$ of virtual classes is identified with an ordered couple $\langle O,P\rangle$ of virtual classes if and only if M is identical with O and N with P. Thus expressions such as

$$'SL_n = SL_m'$$

are already provided for, and two semantical systems may be regarded as identical if and only if their axioms are the same and their truths are the same (and in the same domain of objects). Another kind of semantical equivalence is obtained if we require that L_n ThmEquiv L_m and that the truths of each coincide. And still another kind if we require that L_n FmlaEquiv L_m and that their truths coincide. Without coincidence of their truths in the same domain of objects, it is doubtful if we should wish to regard two semantical systems as equivalent in any strict sense. But suppose we have two domains of objects D_1 and D_2, and suppose '$\mathrm{Tr}^n_{D_1}$' and '$\mathrm{Tr}^m_{D_2}$' to be defined appropriately. Suppose further that D_1 and D_2 are not the same, and hence are either disjoint or overlap. If L_n AxEquiv L_m and also $\mathrm{Tr}^n_{D_1}$ and $\mathrm{Tr}^m_{D_2}$ coincide, we have another important kind of semantical relation to consider. And under similar circumstances but where L_n ThmEquiv L_m or L_n FmlaEquiv L_m. Here also there are various interesting kinds of isomorphism to consider.

These rough and tentative remarks must suffice to point out some of the salient features of *GM*, *GM'*, *GSM*, and *GSM'*.

I. Paradox of Skolem. In closing the discussion of first-order constructivism and its philosophical and methodological justification, we turn to another argument in behalf of first-order, denumerable systems. This argument arises from the situation in the foundations of mathematics known as the *Skolem paradox*. The discovery of this "paradox" is often regarded as one of the most important achievements in modern logic, and it has far-reaching significance for mathematics, philosophy, and the methodology of the sciences. Very little of a non-technical kind has been written on the subject, and we shall therefore give in this closing section a brief account of the nature of this situation, of its significance for mathematics and for the theory of formalized languages, and of its connection with first-order constructivism.

The Skolem paradox is closely related to a generalization of the theorem of Löwenheim discussed in the preceding section. In fact, the importance of Skolem's simpler proofs of the Löwenheim theorem mentioned there is that they lead immediately to this generalization. The generalization states that if the members of a denumerable set of sentences of a language-system of first order are simultaneously satisfiable in any non-null domain whatsoever, then there exists a domain at most denumerable in which they are also simultaneously satisfiable. More specifically, suppose 'K' stands for any denumerable virtual class (or class) of expressions of L_n. Let '$\mathrm{Tr}^n_{D_m}$', as in § H above, symbolize the truth-concept for L_n as defined within *GSM* (or *GSM'*) in terms of '$\mathrm{Cmprh}^n_{D_m}$'. And let D_0 be the (denumerable) domain of natural numbers. Within *GSM* (or *GSM'*) the generalization of Löwenheim's theorem is then to the effect that every formula of the form

$$'(a)(\mathrm{Sent}_n\, a\, .\, K\, a :\supset: \mathrm{Tr}^n_{D_m}\, a)\, .\supset .\, (a)(\mathrm{Sent}_n\, a.\, K\, a :\supset: \mathrm{Tr}^n_{D_0}\, a)'$$

is provable. The generalization itself is for variable 'D_m' and hence can be expressed in *GSM* (or *GSM'*) only as a schema. Skolem has given various proofs of this generalization, some of which utilize an axiom of choice.[1] Perhaps these proofs can be mirrored within *GSM* (or *GSM'*) upon a suitable choice of semantical axioms.

The Löwenheim–Skolem theorem, as this generalization is usually called, has consequences of the utmost importance when applied to systems of axiomatic set theory of first order, such as S above (Chapter VI). The axioms and theorems of such a system are presumably intended to furnish implicitly an analysis of the notion of *set*. A simple enumeration of the infinity of sentences of such a system may easily be set up. Among the set-theoretic axioms or their consequences, there will appear certain statements which, in the usual interpretation, will say that non-denumerably many objects of the theory exist. E.g., one of the theorems of general set theory will state that the cardinality of the set of all real numbers between 0 and 1 is non-denumerable. The derivability of theorems of this kind is, of course, essential if the theory is to be adequate for mathematical analysis. But according to the Löwenheim–Skolem theorem, there must also exist an interpretation of the axioms of set theory within a denumerable domain, e.g., in the domain of the positive integers. In this interpretation the 'ε' of set theory comes to have purely number-theoretic significance. All consequences of the axioms must also hold in this interpretation. Hence the theorem about the cardinality of the real numbers now becomes a theorem concerning

[1] Skolem (1), pp. 10–12; Skolem (2), pp. 220–5; and Skolem (3), pp. 28–9.

the positive integers. The non-denumerable and all theorems concerning it have entirely disappeared. If we attempt, in other words, to characterize the non-denumerable by means of a first-order axiom system, we shall always be led to another interpretation within the denumerable.

This surprising fact is the paradox of Skolem. It is a paradox not in the sense of being a formal contradiction, but only in the loose sense of being a very unusual and unexpected situation.

These considerations constitute also what Skolem calls the *relativity of the concept of set* (*mengentheoretischer Relativismus*). What we are to mean by *set*, and hence by various concepts depending upon this notion, must always be *relative to a given axiom system in a given interpretation*. E.g., 'finite', 'infinite', 'simply infinite sequence', 'power of the continuum', etc., stand for concepts which have only relative meaning. Such a relativity, Skolem notes, is inextricably bound up with every first-order axiom system which purports to be adequate for the classical theory of real numbers.

This situation does not, as we have already remarked, lead to a formal contradiction. An infinite set M can significantly be said to be non-denumerable only in the sense of a particular set of axioms in a particular interpretation. This means roughly that *within the theory* no one-one mapping between M and some denumerable set exists. At the meta-mathematical level, the fundamental domain of the theory as well as (the non-denumerable set) M can be mapped one-to-one with the set of natural numbers and hence with each other. Such a mapping cannot be constructed within the theory but only in the meta-theory.

The Skolem paradox also holds for first-order systems based upon the simplified theory of types, such as T (Chapter VI). The classes, classes of classes, etc., of T are regarded as objects of the fundamental domain, and hence quantification over classes of classes, e.g., is a special case of quantification over the objects of the fundamental domain.

(In the 1929 paper, Skolem gave another proof of set-theoretc relativism without using the generalized Löwenheim theorem. For this proof a somewhat special formulation of axiomatic set theory is needed. In this formulation, which utilizes the Hilbert ε-function (II,F), operations such as forming the power set of a set, of forming the cardinal couple of any two sets, etc., are taken as primitive. Also an operation of bounded abstraction plays an important role, i.e., the operation of forming the set of just those members of some given set which satisfy a given condition. The axioms, which include the *Ersetzungsaxiom* (*S9* of (VI,A)), give substantially the full deductive

power of Zermelo's axioms (including the *Ersetzungsaxiom*). Skolem then sets up a simple enumeration of the totality of terms and formulae of the system. Because the number of terms in such an enumeration is denumerable another interpretation of each term is given within the domain of the natural numbers. Hence the formulae of the theory become satisfiable within a denumerable domain.)

Some very powerful systems have been constructed which purport to avoid this situation. If one is to construct an axiomatic set theory, e.g., which in some sense avoids it, some presupposition of the non-denumerable in the basic logic is presumably required. It might seem that this could be accomplished by presupposing a logical calculus of *second* order as part of the basic logic. But if one regards set theory as an attempt to analyze the notion of set, this would lead to a kind of circularity. The presupposition of a logical calculus of second order is tantamount to presupposing already the notion of set, the very notion we are attempting to analyze. As Skolem has remarked, we must not presuppose in formulating an axiomatic set theory a notion of set itself, but only a meta-mathematical notion of domain ("Bereich"). To presuppose the notion of set in the basic logic would presumably be contrary to the very intent of an axiomatic set theory.

The kind of formulation of simplified type theory provided by T_ω (VI,G) avoids both the Skolem paradox and the kind of circularity mentioned above. But such a system cannot be regarded as of first order—we must in fact regard the whole of T_ω as the basic logic. Nor can such a system be regarded as furnishing an axiomatic characterization of the notions of individual, class, class of classes, etc. These notions are presupposed rather in the underlying logic. It is desirable to keep the underlying logic at a minimum, and it appears to be generally agreed upon that first-order logic ought to provide an adequate basic logic for set theory and hence mathematics, and possibly for scientific and philosophical theories as well. Hence this kind of an interpretation of simplified type theory is not altogether satisfactory, if we regard it as characterizing axiomatically a notion of set.

A third possible kind of system which can be said to avoid the Skolem paradox is to be found in intuitionistic systems or in systems based upon the ramified theory of types.[1] Such theories are severely

[1] See, e.g., A. Heyting, "Die Formalen Regeln der Intuitionistischen Logik", *Sitzungsberichte der Preussischen Akademie der Wissenschaften* (1930, Phys.-math. Klasse), pp. 42–56, 57–71, and 158–69; and, e.g., F. B. Fitch, "The Consistency of the Ramified *Principia*", *The Journal of Symbolic Logic*, 3 (1938), pp. 140–9.

restricted, however, and do not provide a basis for all of classical analysis. In the case of both kinds of systems mentioned above, adequacy for practically the whole of mathematical analysis is achieved. Hence ramified and intuitionistic theories are too restricted for present considerations.

Although "solutions" of these three types exist, none of them can be regarded as wholly acceptable. The Skolem paradox provides a very serious indictment against all known first-order axiom-systems of sufficient power for mathematical analysis. Every known first-order system which is powerful enough either gets into the situation directly or can be interpreted in such a way that it does. And furthermore, as Gentzen remarks, it is by no means "ersichtlich" that a first-order system of sufficient power could be constructed which would avoid it.[1]

We now note the interrelation between this situation and the semantical meta-languages formulated above.

For a non-denumerable L, a translational semantical meta-language of L, howsoever formulated, is subject to this situation. This is evident by observing that the cardinality of the fundamental domain of the whole meta-language must be at least as great as that of L itself. For a denumerable L, a semantical meta-language of the definitional kind (VII,E) is also subject to it. The reason is that objects of higher type and quantification upon them are required for the definition of semantical relations. On the other hand, for denumerable L, SM_D^L and the allied semantical meta-languages formulated in this book avoid the situation. The fundamental domain of such a meta-language is denumerable to begin with. Of course for

[1] G. Gentzen, "Die Gegenwärtige Lage in der Mathematischen Grundlagenforschung", *Forschungen zur Logik und zur Grundlegung der exakten Wissenschaften* (Hirzel, Leipzig, 1938): pp. 1–18; also *Deutsche Mathematik*, 3 (1938), pp. 255–68. For further remarks on the ontological and epistemological significance of the Skolem paradox, by G. D. W. Berry and J. R. Myhill, see the Symposium "On the Ontological Significance of the Löwenheim–Skolem Theorem", *Academic Freedom, Logic, and Religion*, Papers of the American Philosophical Association, Eastern Division (University of Pennsylvania Press, Philadelphia: 1953), pp. 39–70. For important further work related to this situation in one way or another, see, e.g., L. Henkin, "Completeness in the Theory of Types", *The Journal of Symbolic Logic*, 15 (1950), pp. 81–91; J. B. Rosser and Hao Wang, "Non-Standard Models for Formal Logics", *ibid.*, pp. 113–29; Hao Wang, "Arithmetical Models for Formal Systems", *Methodos*, (1951), pp. 217–32, and ' "Arithmetic Translations of Axiom Systems", *Transactions of the American Mathematical Society*, 71 (1951), pp. 283–93. See also J. Kemeny, "Unified Foundations for Semantics", to appear, and Th. Skolem *et al.*, *Mathematical Interpretation of Formal Systems* (North-Holland Publishing Co, Amsterdam: 1955).

non-denumerable L, SM_D^L and the allied meta-languages do get into this situation, and for its avoidance L itself must be denumerable. But still this avoidance for denumerable L provides an important advantage for the translational semantics based upon a special primitive (such as multiple denotation) over the more usual formulations of semantics. For non-translational semantical meta-languages such as SM_C^L, the situation is avoided altogether, its fundamental domain being denumerable whether that of L is or not.

To be sure, one normally expects more than one possible interpretation for any language-system, whether a semantical meta-language or not. The crucial point about the Skolem paradox is that in the two interpretations involved the fundamental domains have different cardinalities. It is this that is avoided by the semantical meta-languages here. Of course Skolem's argument may provide a second interpretation for each of these languages. But the fundamental domain in this second interpretation will also be denumerable.

The existence of the Skolem paradox provides another argument in behalf of the general methodological maxim. Denumerable, first-order language-systems, whether as object-languages or as syntactical or semantical meta-languages, avoid the situation *ab initio*. This situation cannot therefore be regarded as an indictment against such systems. The preference for the denumerable illustrated throughout, and especially in the formulation of non-translational semantics, is thus firmly founded upon the Löwenheim–Skolem theorem.

INDEX OF NAMES

Ackermann, W. 32n, 49, 54n, 58n, 149n, 271.
Aquinas, St. Th. 124n.
Aristotle 123, 124n.

Beierle, J. x.
Bernays, P. 32n, 36n, 49, 53n, 56n, 66, 68, 143, 145n, 163n, 177n, 287n.
Berry, G. D. W. 171n, 294n.
Beth, E. W. x, 2n.
Birkhoff, G. 113n.
Bocheński, I. M. 124n.
Braithwaite, R. B. 2n.
Brouwer, L. E. J. 266.
Burali-Forti, C. 20.

Carnap, R. ix, 16n, 17n, 18, 19, 21, 22, 23, 26, 28, 30, 62n, 64n, 67, 68, 69, 74n, 96n, 119n, 129n, 144n, 166n, 169n, 267, 269, 278, 279.
Chomsky, N. 229n.
Church, A. ix, 27, 32n, 35n, 49, 54n, 58n, 145, 163n, 166n.
Chwistek, L. 27, 71, 151n, 264.
Clarke, F. x.
Copi, I. 32n.
Couturat, L. 113n.

DeMorgan, A. 40.

Epimenides 20, 135.
Eubulides 20.

Fitch, F. B. x, 16n, 18n, 55n, 69, 138n, 226n, 293n.
Fraenkel, A. 145.
Frege, G. ix, 14, 36, 267.

Gentzen, G. 271, 294n.
Gödel, K. 73, 135, 136, 137, 138, 139, 141, 145n, 226.
Goodman, N. ix, 18n, 29, 65, 68, 69, 188n, 227n, 229, 237n, 266n, 267, 268, 269, 271.
Götlind, E. 36n.

Harris, Z. 21n.
Hempel, C. G. x, 16n.
Henkin, L. 294n.
Heyting, A. 293n.
Hilbert, D. 32n, 49, 53n, 54, 56, 58n, 66, 68, 143, 163n, 177n, 195, 260, 287n, 292.
Hiz, H. x.
Hobbes, T. 26, 104n, 105n, 202n.
Hull, C. 16n, 69.
Huntington, E. V. 143.
Hutten, E. 2n.

Jevons, W. S. 26

Kaplan, A. 123n.
Kemeny, J. 144n, 156n, 175n, 266n, 294n.
Keynes, J. N. 26.
Kleene, S. C. 32n.
Kuratowski, C. 155, 162, 163.

Leblanc, H. 32n.
Leśniewski, S. 65.
Lewis, C. I. 68.
Löwenheim, L. 286, 287, 291, 292, 295.

McKinsey, J. C. C. 16n.
MacLane, S. 113n.
Meager, R. x.
Mill, J. S. 26.
Morris, C. 21n.
Morrow, G. R. x.
Myhill, J. 226n, 294n.

Nagel, E. x.
Neumann, J. von 145n.

Ockham, Wm. of 105n.
Ogden, C. K. 63n.

Peano, G. 14, 50, 67
Peirce, C. S. 64.
Plato 14.
Price, H. H. ix.

Quine, W. V. ix, 27, 29, 32n, 35n, 51, 60, 65, 71, 79n, 82, 141, 145n, 196n, 224n, 227n, 229, 237n, 264, 267n, 268, 269, 271, 281n.

Ramsey, F. 20n, 151n.
Rasiowa, H. 36n.
Reichenbach, H. 16n.
Rieman, G. F. x.
Robbins, B. L. x.
Rosser, B. 32n, 139n, 141n, 294n.
Russell, B. ix, 5n, 9, 14, 15, 20, 26, 55, 56n, 143, 144, 151, 152, 154, 204n, 213, 229n, 267.
Ryle, G. 24n.

Savage, T. R. x.
Schmidt, A. 61n.
Schrecker, P. x, 20n.
Schröter, K. 282n.
Skolem, T. 145, 191, 270 f., 287, 290, 291, 292, 293, 294, 295.
Sugar, J. 16n.
Suppes, P. 16n.

Tarski, A. ix, 18, 27, 28, 30, 64n, 71, 73, 74n, 101n, 107n, 109n, 113n, 119n, 123, 124n, 133n, 137, 138n, 140n, 143, 144n, 151n, 153n, 172n, 173n, 174n, 177n, 186n, 189n, 225, 226, 264, 269, 282n, 287n.
Taylor, A. E. 14.

Wang, H. 2n, 61n, 144n, 149n, 175n, 294n.
Wessel, A. 234n.
Whitehead, A. N. v, 5n, 12, 14, 56n, 151n, 204n, 267.
Wiener, N. 155, 162, 163.
Woodger, J. H. x, 2n, 9, 13, 14n, 16n, 69, 74n, 105n, 227n, 245n.

Zermelo, E. 15, 18, 28, 61, 143, 144, 145, 149, 191, 293.

INDEX OF SUBJECTS

Abstraction, operation of 49 ff., 81, 157, 214 f., 218 f., 230, 239, 288. *See also* Rule(s) *and* Theorem(s).
Abstraction in science 21 ff.
Adequacy of a truth-definition 119 ff., 183 f., 249 ff. *See also* Rule(s).
Assertion sign 36.
'Axiom of *L*' 86 ff., 242, 284.
'Axiom of *S*' 149 ff.
'Axiom of *T*' 156 ff.
Axioms, descriptive 54, 88, 242.
  logical 36 f., 51, 53, 88, 242. *See also* Rule(s) *and names of specific axioms and systems.*

Bold-face letters, use of 34 f.
Boolean algebra 112 ff., 189, 192 ff.

Character, typographical 233.
Choice, Axioms of 147, 150, 153.
Classes 65, 151 f., 272 f.
  virtual 51, 106 f., 272 f.
Closure of a formula 90.
Co-extensiveness 104, 181, 246, 255.
Common predicate constants 105, 182, 246.
Completeness 11 f., 97, 131, 139 f.
Comprehension 104, 180, 188 f., 246, 254, 279 f., 285.
Concatenation 72 ff., 231 ff., 262.
Conditional, material 35, 81, 241.
Conjunction 35, 81, 240.
Conjuncts 129.
Connectives 32.
Consequence 3, 10.
  immediate 81, 284.
Consistency 11, 25, 28, 29, 97, 125 ff., 159, 206, 257, 265 f.
  relative 132 ff., 206 ff., 257.
Constants 3.
  functional 4, 29, 31, 33, 57, 222 f.
  individual 4, 29, 31, 33, 57, 214 ff., 257 f.
  logical 4, 31, 89.
  predicate 4, 31, 80, 89, 158.
  (defined) predicate 51.

Decidability 139 ff.
Definiendum 5.
Definiens 5.
Definition, abbreviatory 3, 6, 12.
  recursive 4, 34, 71, 174 f.
Denotation, multiple 99 ff., 107, 172, 219, 245.
Denoting expressions 26, 181, 187, 255. *See also* Rule(s).
Denumerabilism 270 ff.
Derivability 96.
Descriptions, selective 55 f.
  singular 56 f.
  structural 73 f.
'Descriptive Theorem' 89.
Designation 166 ff., 170 ff.
Determination 169 ff.
Dialectic 14.
Disjunction 32. *See also* Rule(s).
Disjuncts 129.
Domain of individuals 60 ff., 285 ff.

Economy 265 f.
Empiricism 19.
Equivalence, between systems 289 f.
  deductive 97.
  material 35, 81, 129.
Exclusiveness 129.
Existence of Classes, Axiom of 153.
Explication 19 f.
Expressions 3, 65, 78. *See also* Rule(s).
Extension of a predicate constant 106, 246.
Extensionality, Axioms of 146, 150, 153.

Falsity 129.
Finitism 266 ff.
Formalization 2 ff.
Formula 4 f., 33, 62, 80 ff., 146, 237 ff. *See also names of specific systems.*
Foundation, Axiom of 148, 150.
Framed ingredients, method of 82 ff. *See also* Rule(s).
Free variables, *see* Variables, free.

Generalization, Rule of, *see* Rule(s).
GM 283 ff.
GM' 285 ff.
GSM 286 ff.
GSM' 289 f.

Identity 51, 52 ff., 148, 154, 232. *See also* Rule(s) *and* Theorem(s).
IM 230 ff.
Implexion, *see* Semantical implexion.
Implication, material 129.
Individuals 31.
  in narrow sense 60 f.
  in wider sense 60 f.
  individuals_1 162.
  individuals_2 162.
Infinity, Axiom of 147 f., 150, 153.
Inscriptions 29, 64 ff., 73, 227 ff., 254 ff., 275 ff.
Interpretation 6 ff.
  non-standard 290 ff.
  partial 280.
  standard 7.
ISM_C^L 254 ff.
ISM_D^L 245 ff.
Isomorphism between systems 289 f.

L 53 f.
*L** 131 f.
Languages, natural 1, 13, 20 f.
Language-systems 2 ff., 285 ff.
  fully formalized 17.
  of finite order 161 ff.
  of first order 27, 31 ff., 61, 66 ff.
  partially formalized 17 f.
  simple, applied 54.
'Logical Axiom of *L'* 88, 242.
'Logical Theorem of *L'* 88 f., 243.
Logistic systems 2 ff., 282 ff.

M 71 ff.
Mathematical induction 67.
Maxims, methodological 61, 263 ff., 269.
Membership 146, 152.
Meta-axiom, *see* Rule(s).
Meta-languages 10, 62 ff.
  semantical 64, 99 ff., 215 ff., 245 ff.
  syntactical 63, 70 ff., 214 f., 229 ff.
Meta-mathematics 15.
Meta-theorems, *see* Rule(s) *and* Theorem(s).
Minimal semantics 24 ff., 179 ff., 254 ff., 276 ff.
Modus Ponens, *see* Rule(s).
Multiplicative Axioms 154.

Names, *see* Constants *and* Proper names.
Negation 32. *See also* Rule(s).
Nominalism 29, 65, 266 ff.
Normal forms 57 ff., 116 ff., 119 ff.
Null predicate constants 106, 181, 246.
Null Set, Axiom of Existence of 146, 150.

Object-language 10, 62 f.
Occurrence of an expression 85. *See also* Variables.
Ordered pairs 155, 288.

Pairing, Axiom of 146, 150.
Paradoxes 20 f.
  of Burali-Forti 20.
  of Epimenides 20.
  of Russell 9, 20, 152.
  of Skolem 290 ff.
Peano's postulates 67.
Pedagogy 13.
Power Axiom 147, 150.
Proof 5, 38, 76, 96, 242 ff.
Proper names 216, 223, 257 ff.

Quale 68.
Quantifiers 31, 35, 47 f. *See also* Rule(s) *and* Theorem(s).
Quasi-formula 239.
Quasi-quotation 34 f.
Quotation marks 34 f.
  formal theory of 258 ff.

Range of a variable 105, 160 f.
Rational reconstruction 19 f.
Rationality 14 f.
Realism 268 ff.
Recursion equations 67.
Replacement, Axiom of 148, 150. *See also* Rule(s).
Re-interpretation 160 ff.
Rigor, mathematical 15.
Rule(s) 36.
  of Abstraction 51, 88, 153.
  of Adequacy 119, 121, 251, 253.

Rule(s)—(*contd*).
of Comprehension 184 ff.
of Consistency 128, 206, 253, 257.
of Denotation 109, 157 f., 249.
of Denoting Expressions 185.
derived 39.
of Designation 167 f.
of Determination 170.
of Differentiation 186.
Disjunction 122.
of Distinctness 75, 157 f.
of Existence 234.
of Expressions 95.
of Framed Ingredients 92 f.
of Generalization 37, 80 f., 153, 199.
of Identity 53, 75, 87.
of Inference 5, 37.
of Infinite Induction 75.
of Modus Ponens 37, 81.
Negation 122, 185.
of Non-Null Semantical Products 186.
Product 185.
of Quantification 37, 87, 108 f., 153.
of Replacement 43 f.
of Restriction 109, 158, 249.
of Satisfaction 165 f.
of Semantically Null Products 186.
of Shapes 234 f.
Sum 185.
Truth-functional 36, 86.
of Vacuous Abstracts 186 f.
See also Theorem(s) *and names of specific systems.*

S 145 ff.
Satisfaction 28, 107, 165 f., 171 f., 174 f. *See also* Rule(s).
Satisfiability 286 f.
Self-reference 135 ff.
and comprehension 223 ff.
Semantic(al) constructivism 266 ff.
discreteness 188.
implexion 195 ff.
negatives 108, 183, 246.
overlapping 188.
products 108, 183, 246.
sums 107, 183, 246.
systems, formal theory of 281 ff.
Semantics 10, 63 f.
definitional 173 ff.
non-translational 28, 179 ff., 273 ff.
translational 28, 99 f., 273 ff.
Sentences 86.
atomic universal 58, 89.
universal 90.
Sentential functions 4, 86, 89.
Set theory 145 ff.
Shapes 29, 64 ff., 73, 247. *See also* Rule(s).
Sign-designs, *see* Shapes.
Sign-events, *see* Inscriptions.
SM_O^L 180 ff.
$SM_O^{L'}$ 213 ff., 219 f.
SM_D^L 101 ff., 175 ff.
$SM_D^{L'}$ 213 ff.
SM_D^{L+} 138.
SM_O^S 203 ff.
SM_D^S 149 ff.
SM_D^T 156 ff.
Specific Hypotheses of Comprehension 203 ff., 225.
SSM^C 224 ff.
Subject-matter 8.
Subset Formation, Axioms of 148, 150.
Subsumption 104, 181, 246.
Sums, Axiom of 147, 150.
Syntax 10, 63 f.
first order 70 ff., 149 ff., 156 f., 214 f., 229 ff., 257, 283 ff.

T 151 ff.
T_ω 162 ff.
Terms 4, 33, 62, 214, 223.
syntactical 103.
Theorem(s) 3, 5, 38. *See also* Rule(s).
of abstraction 52.
of concatenation 76 ff., 236, 245.
of descriptions 56 f.
of designation 168 f.
of determination 170.
of identity 53.
of individual constants 216, 220 ff., 258.
of Löwenheim 286 ff.
of proper names 216 ff., 258, 260 f.
of quantification 45 ff.
of semantics 110 ff., 125 ff., 192 ff., 205 ff., 220 ff., 249, 256, 260 f.
of Skolem 291.
of syntax 76 ff., 91 ff., 235 ff., 244 f.

'Theorem of *L*' 88 ff., 240 ff.
Transfinite types 175.
True shapes 247 f.
Truth 5, 11, 18, 115 ff., 158 f., 166, 169, 183, 217 f., 220, 246, 251 f.
Types, simplified theory of 151 ff.

Undecidability 139 ff.
Understanding a language-system 278 ff.
Unit predicate constants 105, 182, 246.
Universal predicate constants 106, 182, 189 f., 246.
Univocality, principle of 26 f.

Vacuous abstracts and truth 251 ff.
Variables 3, 31, 71, 80, 153, 156.
  bound 37, 51, 85.
  free 37, 51, 84 f., 241.
Virtual classes, *see* Classes, virtual.
Vocabulary, descriptive 14.
  logical 14.
  primitive 2, 3, 32.

INDEX OF FORMAL META-THEORETIC SIGNS

Abst 81, 157, 214, 239.
AbstAx 88, 215, 242.
ac 72.
Ac 230.
AC 224.
AcString 79, 237.
AtFmla 80, 157, 214, 237, 283.
AtFmla_n 284.
AtUnivSent 89, 244.
Ax 88, 157, 242.
Ax_n 283.
AxEquiv 289.
ay 214, 224.
Ay 257.
Ay′, etc. 257.
AY 224.

Bgn 79, 237, 283.
BOcc_c 85.

C 231, 234.
C_s 240.
Char 233.
ChoiceAx 150.
Clsr 90, 244.
Clsr′ 90, 244.
cmp 224.
CMP 224.
Cmplt 97.
Cmprh 104, 180, 188, 189, 246, 254, 280, 297.
$\mathrm{Cmprh}^n_{D_m}$ 285.
Cmprh′ 207.
Cmprh^{oo} 220.
cnc 224.
CNC 224.
CNF 117.
Cnst 97.
CoExt 104, 181, 246, 255.
CoExt^{oo} 220.
Com 105, 182, 246.
Conj 117, 129.

D 238.
D_s 239.
DE 181, 187, 255.
DE^{oo} 220.
DedEquiv 97.
Den 100, 107, 172, 219, 246.
Des 167, 170, 172.
DesAx 88.
DesThm 89.
Det 170, 171.
Disj 117, 129.
dot 81.
Drv 96.

eff 222.
Ends 79, 237, 283.
EqLng 238.
EqPar 239.
Equiv 129.
EvStr 252.
EvTildeStr 252.
EvTildeString 174.
ex 72.
Ex 231.
Excl 129.
exisqu 81.
Exp 78.
Ext 106, 246.
ExtAx 150.

FD*L*1 131.
FD*L*2 131.
Fls 129.
Fmla 84, 215, 240, 283.
Fmla_n 284.
FmlaEquiv 289.
FoundAx 150.
FrIng 83, 90, 283.
FrOcc_c 85.
FuncCon 223.
FV 85, 90, 241.

Gen 80, 242.

hrsh 81.

IC 81, 242.
IC_n 284.
id 72.
Id 230.
ID 224.
IdAx1–2 88, 215, 242.
Imp 129, 196.
InCon 214.
InfAx 150.
invep 72.
INVEP 224.

L_n 289.
Like 238.
Lngr 232 f.
LogAx 88, 242.
LogThm 89, 244.
lp 72.
Lp 230.
LP 224.

MC 241.
MinConj 117.
MinDisj 117.
MP 81, 242.

N 238.
Neg 108, 183, 246.
Neg′ 209.
Null 106, 181, 189, 246.
Null′ 209.
NullSetAx 150.

O 188.
Occ 85.
Occ_c 85.
OddTildeStr 252.
OddTildeString 174.

PairAx 150.
Part 232.
pee 72.
Pee 231.

PowerAx 150.
PNF 118.
Pr_b 84, 90.
PredCon 89, 244, 283.
$PredCon^{n+1}$ 158.
PrimPredCon 80, 237, 283.
$PrimPredCon_n$ 283.
PrNm 216, 223, 258.
Prod 108, 183, 246.
Prod′ 209.
Proof 96, 243.
PrSentFunc 86.
PS 76, 156, 284.
PS_n 284.
PsAtSat 174.

qu 81.
Quant 238.
QuantAx1–2 87, 215, 242.
QuantString 90, 244.
QuasiFmla 239.

ReplAx 150.
R^{invep}_{rp} 116.
rp 72.
Rp 230.
RP 224.

S^b_c 215.
S'^b_c 215.
$S1^b_c$ 215.
$S1'^b_c$ 215.
$S?^b_c$ 215.
$S?'^b_c$ 215.
Sat 107, 165, 171, 174.
Seg 79, 237.
SemCmplt 131.
Sent 86, 244.
SentFuncOne 89, 244.
SentFuncTwo 89.
SetAx 151.
SF^b_c 87, 157, 241.
$SF1^b_c$ 87, 157.
$SF?^b_c$ 87, 91, 157.
SHC 205, 225.
SL_n 288.
SmTp 157.
Step 242.
Sub 104, 181, 246.
Sub^o 217.
Sub^{oo} 217.
subscr 156.
SubscrString 156.
SubsetFormAx 150.
Sum 107, 183, 246.
Sum′ 208.
SumAx 150.

TFAx1–4 86, 241.
Thm 89, 151, 214, 243.
Thm′ 207.
ThmEquiv 289.
tilde 72, 81.
Tilde 230.
TILDE 224.
TildeStr 252.
Tr 118, 119, 159, 169, 183, 189, 218, 220, 253, 256.
Tr_1 116, 220, 246, 251.
Tr_2 118, 220, 252.
Tr_3 217, 220, 252.
Tr_4 252.
Tr_5 253.
Tr^n_{Dm} 287.
Tr′, etc. 158.
tripbar 81.
Trm 214, 223.
TS 248.
TS_1 247.
TS_2 247 f.

Unit 105, 182, 246.
$Unit^{oo}$ 220.
Univ 106, 182, 189, 190, 246.
Univ′, etc. 158, 209.
UnivSent 90, 244.
US 139.

VacAbst 251.
$ValVbl_L$ 161.
Vbl 80, 156, 237.
Vbl_1 or Vbl′ 158.
Vbl_2 or Vbl″ 158.
VblString 284.
vee 72, 81.
Vee 230.
VEE 224.

SPECIAL SYMBOLS

v 32.
$\sim$ 32.
x, x', etc 32.
P, P′, etc. 32.
ā, ā′, etc. 33.
ī, ī′, etc. 33.
$\boldsymbol{A}$, $\boldsymbol{B}$, etc. 34, 108.
$\boldsymbol{x}$, $\boldsymbol{y}$, etc. 34, 108.
$\supset$ 35.
. 35.
$\equiv$ 35.
(E$\boldsymbol{x}$) 35.
$\vdash$ 36.
ε 49, 146, 152, 161 f., 288.
$\ni$ 49, 71, 102, 288.
$\boldsymbol{F}$, $\boldsymbol{G}$, etc. 51, 75.
$=$ 51, 52, 148, 152, 154, 232.
$(\varepsilon \boldsymbol{x}.\boldsymbol{F}\boldsymbol{x})$ 55, 195.
$(\iota \boldsymbol{x}.\boldsymbol{F}\boldsymbol{x})$ 56.
E! 56, 182.
$<$ 66.
o 66.
σ 66.
a, b, etc. 71.
$\cap$ 72.
$\#$ 74.
$\subset$ 147, 154.
$x^{\boldsymbol{n}}_{\boldsymbol{m}}$ 153
$\langle x^1_1, x^1_2 \rangle$ 155.
OP 155.
F, G, etc. 173.
R, Q, etc. 173.
$\int$ 188.
$(a + b)$ 195.
$(a . b)$ 195.
$-a$ 195.
$\boldsymbol{S}$ 234.
$(\boldsymbol{S}_1 \ldots \boldsymbol{S}_n)$ 235.
$\boldsymbol{X}$ 235.
$\boldsymbol{A}$, etc. 247.
$\boldsymbol{x}$, etc. 249.
$\boldsymbol{a}$ 257.
$\boldsymbol{Ay}$ 257.
$\boldsymbol{s}_1$, etc. 259.
$'\boldsymbol{s}_1 \ldots \boldsymbol{s}_n'$ 259.
$\cdot\boldsymbol{s}_1 \ldots \boldsymbol{s}_n\cdot$ 260.
$\boldsymbol{F}^n_m \boldsymbol{F}^q_k \ni \boldsymbol{X}$ 288.
$\langle \mathrm{Ax}_n, \mathrm{Tr}^n_D \rangle$ 288.

UNIVERSITY COLLEGE
Date Due
COLLEGE LIBRARY

Due	Returned	Due	Returned
APR 0 1 2001			
	MAR 0 5 2[illegible]		

C.41

Truth & denotation; main
164M382t C.2

3 1262 03264 2802

164
M382t
c.2